Calhoun
Institutional Archive of the Naval Postgraduate School

Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations | 1. Thesis and Dissertation Collection, all items

1972

Mathematical models and computer solution for the equations of motion of surface ships and submarines, in six degrees of freedom.

Romero, Edgardo Juan.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/16244

Downloaded from NPS Archive: Calhoun

DUDLEY KNOX LIBRARY

http://www.nps.edu/library

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

MATHEMATICAL MODELS AND COMPUTERS SOLUTION FOR THE EQUATIONS OF MOTION OF SURFACE SHIPS AND SUBMARINES, IN SIX DEGREES OF FREEDOM

Edgar Romero

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

MATHEMATICAL MODELS AND COMPUTER SOLUTION FOR THE EQUATIONS OF MOTION OF SURFACE SHIPS AND SUBMARINES, IN SIX DEGREES OF FREEDOM

by

Edgar Romero

Thesis Advisor: G.J. Thaler

June 1972

T147793

Approved for public release; distribution unlimited.

Mathematical Models and Computer Solution for the Equations of Motion of Surface Ships and Submarines, in Six Degrees of Freedom

by

Edgar Romero
Lieutenant (junior grade), Colombian Navy
Colombian Naval Academy, 1965
B.S., Naval Postgraduate School, 1972

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

ABSTRACT

From the standard equations of motion, mathematical models are developed and standardized for simulation in analog or digital computers. A complete study is done of simulation techniques as applied to the mathematical models. A new technique for simulation of very complex systems of equations including nonlinearities is developed. The model can be used in Control studies. Digital simulations of the dynamics of the semisubmerged ship and a submarine in six degrees of freedom are done. This latter with all the nonlinearities and cross-coupling terms included.

TABLE OF CONTENTS

I. THE EQUATIONS OF MOTION ---------- 11

A. DERIVATION ---------- 11

B. EXAMPLE ---------- 19

II. MODELING TECHNIQUE ---------- 26

A. DEVELOPMENT AND STANDARDIZATION ---------- 26

B. NON LINEAR EQUATIONS ---------- 35

C. EXTENSIONS ---------- 37

D. CONCLUSIONS ---------- 42

III. SIMULATION CONSIDERATIONS ---------- 44

A. STABILITY ---------- 44

B. REDUNDANT INTEGRATORS ---------- 46

C. ALGEBRAIC LOOPS ---------- 48

IV. A NEW TECHNIQUE FOR MODELING ---------- 54

V. COMPUTER SIMULATION ---------- 65

A. DEVELOPMENT OF THE COMPLETE PROGRAM ---------- 65

B. EXAMPLE WITH THE TWIN HULLED SEMISUBMERGED SHIP (S^3) ---------- 72

VI. THE SUBMARINE IN SIX DEGREES OF FREEDOM ---------- 81

A. DERIVATION OF MASS AND INERTIAS ---------- 81

B. THE EQUATIONS OF MOTION OF THE SUBMARINE ----- 83

VII. CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK - 99

COMPUTER PROGRAMS ---------- 101

1 USING MEMORY ---------- 101

2 USING IMPLICIT ---------- 102

3 LATERAL DYNAMICS OF THE S^3 ---------- 103

4A LATERAL AND LONGITUDINAL DYNAMICS OF THE S^3 ----- 106

4B TRANSFER FUNCTION ---------- 113

5 SUBMARINE SIX DEGREES OF FREEDOM ---------- 114

6 PROGRAM EXPLANATION WITHOUT DATA ---------- 123

7 SUBROUTINE TO CALL DRAW ---------- 128

LIST OF REFERENCES ---------- 130

INITIAL DISTRIBUTION LIST ---------- 131

FORM DD 1473 ---------- 133

LIST OF TABLES

I MAXIMUM LOOP GAIN AND NUMBER OF ELEMENTS -------- 51

II COMPARISON OF RESULTS ---------- 76

LIST OF DRAWINGS

1-1 Coordinate system fixed in vehicle ---------- 11

1-2 Rotation in Pitch ---------- 13

1-3 Moment Relations ---------- 15

2-1 Block diagram Eq. (2-1b) ---------- 26

2-2 Block diagram Eq. (2-1b) ---------- 27

2-3 Block diagram Eq. (2-2) ---------- 27

2-4a Block diagram Eq. (2-5a) ---------- 28

2-4b Block diagram Eq. (2-5b) ---------- 29

2-5 Block diagram Eq. (2-5c) ---------- 30

2-6 Block diagram Eq. (2-6b) ---------- 32

2-7 Summer-Node Representation ---------- 32

2-8 Summer-Node Representation (2) ---------- 33

2-9 Block, Terms in Diagonal ---------- 34

2-10 Block, Terms above Diagonal ---------- 34

2-11 Block, Terms below Diagonal ---------- 34

2-12 Block diagram Eq. (2-7) ---------- 36

2-13 Block diagram Eq. (2-8) ---------- 38

2-14 Block diagram Eq. (2-9) ---------- 38

2-15 Computer solution of Eq. (2-9) ---------- 39

2-16a Block diagram Eq. (2-11) ---------- 40

2-16b Block transformation of Fig. 2-16a ---------- 40

2-17 Block diagram Eq. (2-12) ---------- 41

3-1a Analog diagram Eqs. (3-1) and (3-2) ---------- 45

3-1b Stable case ---------- 46

3-1c Unstable case ---------- 46

3-2 Block diagram Eq. (2-9) ---------- 47

3-3a Block Transformation of Fig. 3-2 ---------- 48

3-3b Constant Multiplier ---------- 49

3-3c Two Multipliers in a loop ---------- 49

3-4 Longitudinal Dynamics of the Semisubmerged Ship (Eq. 1-8) ---------- 53

4-1 Block diagram Eq. (4-10) ---------- 56

4-2 Complete block diagram Eq. (4-10) ---------- 58

4-3 Block diagram Eq. (4-12a) ---------- 59

4-4 Block diagram Eq. (4-12b) ---------- 60

4-5 Block diagram Eq. (4-12c) ---------- 61

4-6 Block diagram for System of Three Equations (Eq. 4-13) ---------- 63

5-1 S^3 Yaw response, U=1.0, δ_r=0.1 ---------- 77

5-2 S^3 Roll response, U=1.0, δ_r=0.1 ---------- 78

5-3 S^3 Heave response, U=1.0, δ_c=0.1 ---------- 79

5-4 S^3 Pitch response, U=1.0, δ_c=0.1 ---------- 80

6-1 Positive directions of axes, angles, velocity components, forces and moments ---------- 87

6-2 Submarine Yaw response, δ_r=-0.1, u(0)=6, u_c=6 --- 88

6-3 Submarine Roll response, δ_r=-0.1, u(0)=6, u_c=6 -- 89

6-4 Submarine Pitch response, δ_r=-0.1, u(0)=6, u_c=6 - 90

6-5 Submarine u response, δ_r=-0.1, u(0)=6, u_c=6 ----- 91

6-6 Submarine v response, δ_r=-0.1, u(0)=6, u_c=6 ----- 92

6-7 Submarine w response, δ_r=-0.1, u(0)=6, u_c=6 ----- 93

6-8 Submarine p response, δ_r=-0.1, u(0)=6, u_c=6 ----- 94

6-9 Submarine q response, δ_r=-0.1, u(0)=6, u_c=6 ----- 95

6-10 Submarine r response, δ_r=-0.1, u(0)=6, u_c=6 ------ 96

6-11 Submarine Pitch response, δ_b=+0.1, u(0)=6, u_c=6 -- 97

6-12 Submarine Pitch response, δ_s=+0.1, u(0)=6, u_c=6 -- 98

TABLE OF SYMBOLS

$\vec{R}$	x,y,z	Vector distance of a point from the origin, and the components along the X, Y, and Z axis $\vec{R} = \hat{i}x + \hat{j}y + \hat{k}z$
$\vec{R}_G$	x_G, y_G, z_G	Vector distance of the center of gravity from the origin $\vec{R}_G = \hat{i}x_G + \hat{j}y_G + \hat{k}z_G$
$\vec{U}$	u,v,w	Velocity of the origin $\vec{U} = \hat{i}u + \hat{j}v + \hat{k}w$
$\vec{\Omega}$	p,q,r	Angular velocity of the body about the origin $\vec{\Omega} = \hat{i}p + \hat{j}q + \hat{k}r$
I_X, I_Y, I_Z		Moment of inertia of the body about the X, Y, Z axis
$\vec{F}$	X,Y,Z	Force on the body $\vec{F} = \hat{i}X + \hat{j}Y + \hat{k}Z$
$\vec{M}$	K,M,N	Moment about origin $\vec{M} = \hat{i}K + \hat{j}M + \hat{k}N$
X_o, Y_o, Z_o		Fixed coordinate system, origin at the surface of liquid -- components of translational position of body
ϕ, θ, ψ		Angular rotation roll, pitch and yaw, e.g., θ angle that longitudinal X axis makes with horizontal plane
δ		Deflection of a control surface
$X_{\square}\ldots$ $M_{\square}\ldots$		Hydrodynamic coefficients
ρ		Mass density of water
g		Acceleration constant, gravity
CSMP		Continuous System Modeling Program
DSL		Digital Simulation Language
C G		Center of Gravity

ACKNOWLEGEMENT

The author wishes to express his sincere appreciation to Dr. George J. Thaler for the guidance, assistance and continuous encouragement which he provided during the pursuit of this study. The author would also like to express his appreciation to Dr. Grant R. Hagen from the Naval Ship Research and Development Center for his valuable assistance in the solution of some of the problems and his kindness in furnishing the data for the submarine simulation.

I. THE EQUATIONS OF MOTION

In this section the equations of motion are developed. Similar developments can be found in many books and the following is just a summary of the work done by Martin A. Abkowitz [Ref. 1]. In the last pages of this section, a specific example is taken: The "Twin hulled semisubmerged ship" and its transfer function is obtained.

A. DERIVATION

Six degrees of freedom: translation along three orthogonal axes and rotation about each one:

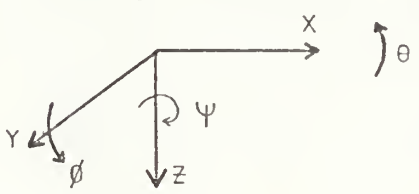


Figure 1-1 Coordinate system fixed in vehicle

Newton's law of motion:

$\vec{F}$ = Force on Body = $\frac{d}{dt}$ ($\overrightarrow{\text{momentum}}$)

$\vec{M}$ = Moment acting on a body = $\frac{d}{dt}$ ($\overrightarrow{\text{Angular Momentum}}$)

$\vec{F} = \frac{d}{dt}(m\vec{U}_G)$ where $\vec{U}_G$ is velocity of center of gravity

$\vec{M} = \frac{d}{dt}(\vec{I\Omega})_G$

For an origin not at the center of gravity but fixed in the vehicle:

$$\vec{U}_G = \vec{U}_o + \vec{\Omega} \times \vec{R}_G$$

$$\vec{U}_o = \vec{U} + \vec{\Omega}_e \times \vec{R}_b$$

where $\vec{U}_o$ is the velocity of the origin

$\vec{U}$ is geographical velocity of the body

$\vec{\Omega}_e$ is angular velocity of earth

$\vec{R}_b$ is radius vector from center of earth to center of vehicle.

$$\vec{U}_G = \vec{U} + \vec{\Omega}_e \times \vec{R}_b + \vec{\Omega} \times \vec{R}_G$$

$$\vec{F} = m \frac{d\vec{U}_G}{dt} = m \frac{d}{dt} (\vec{U} + \vec{\Omega}_e \times \vec{R}_b + \vec{\Omega} \times \vec{R}_G)$$

$$= m \frac{d}{dt} (\vec{U} + \vec{\Omega} \times \vec{R}_G) + m\vec{\Omega}_e \times \frac{d\vec{R}_b}{dt} + \frac{d}{dt} \vec{\Omega}_e \times \vec{R}_b$$

$$\frac{d\vec{R}_b}{dt} = \vec{U}_o \quad , \quad \frac{d\vec{\Omega}_e}{dt} = 0$$

$$\vec{F} = m \frac{d}{dt} (\vec{U} + \vec{\Omega} \times \vec{R}_G) + m(\vec{\Omega}_e \times \vec{U} + \vec{\Omega}_e \times \vec{\Omega}_e \times \vec{R}_G)$$

The term $m\vec{\Omega}_e \times \vec{U}$ (Coriolis force), and $m\vec{\Omega}_e \times \vec{\Omega}_e \times \vec{R}_b$ (centripetal acceleration due to rotation of the earth), are zero as compared with other forces.

$$\vec{F} = m \frac{d}{dt} (\vec{U} + \vec{\Omega} \times \vec{R}_G) \qquad (1\text{-}1)$$

Finding the derivatives of unit vectors (change in direction):

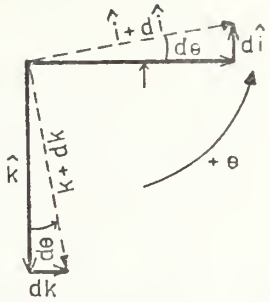


Figure 1-2 Rotation in Pitch

where $d\hat{i} = -\hat{k}d\theta$

$d\hat{j} = 0$

$d\hat{k} = \hat{i}d\theta$

it can also be shown:

$d\hat{i} = \hat{j}d\psi$ $\qquad$ $d\hat{i} = 0$

$d\hat{j} = -\hat{i}d\psi$ $\qquad$ $d\hat{j} = \hat{k}d\phi$

$d\hat{k} = 0$ $\qquad$ $d\hat{k} = -\hat{j}d\phi$

Adding the contributions:

$$\frac{d\hat{i}}{dt} = \hat{i} \cdot 0 + \hat{j}\frac{d\psi}{dt} - \hat{k}\frac{d\theta}{dt}$$

$$\frac{d\hat{j}}{dt} = -\hat{i}\frac{d\psi}{dt} + \hat{j} \cdot 0 + \hat{k}\frac{d\phi}{dt}$$

$$\frac{d\hat{k}}{dt} = \hat{i}\frac{d\theta}{dt} - \hat{j}\frac{d\phi}{dt} + \hat{k} \cdot 0$$

$$\vec{\Omega} = \hat{i}p + \hat{j}q + \hat{k}r \qquad p = \frac{d\phi}{dt}\ ,\ q = \frac{d\theta}{dt}\ ,$$

$$r = \frac{d\psi}{dt}\ ,\ \text{(linear)}$$

$$\frac{d\hat{i}}{dt} = \vec{\Omega} \times \hat{i} = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ p & q & r \\ 1 & 0 & 0 \end{vmatrix}$$

$$\vec{U} = \hat{i}u + \hat{j}v + \hat{k}w$$

$$\vec{R}_G = \hat{i}x_G + \hat{j}y_G + \hat{k}z_G$$

Using the above results, Eq. (1-1) becomes:

$$\vec{F} = m\frac{d}{dt}(\vec{U} + \vec{\Omega} \times \vec{R}_G) = m\left(\frac{d\vec{U}}{dt} + \frac{d}{dt}\vec{\Omega} \times \vec{R}_G\right)$$

$$\frac{d\vec{U}}{dt} = \hat{i}\frac{du}{dt} + u\frac{d\hat{i}}{dt} + \hat{j}\frac{dv}{dt} + v\frac{d\hat{j}}{dt} + \hat{k}\frac{dw}{dt} + w\frac{d\hat{k}}{dt}$$

$$= \hat{i}\dot{u} + u(\hat{j}r - \hat{k}q) + \hat{j}\dot{v} + v(-\hat{i}r + \hat{k}p) + \hat{k}\dot{w} + w(\hat{i}q - \hat{j}p)$$

$$\vec{\Omega} \times \vec{R}_G = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ p & q & r \\ x_G & y_G & z_G \end{vmatrix} = \hat{i}(qz_G - ry_G) + \hat{j}(rx_G - z_Gp) + \hat{k}(py_G - qx_G)$$

$$\frac{d}{dt}(\hat{i}(qz_G - ry_G)) = \hat{i}(z_G\dot{q} - y_G\dot{r}) + \hat{j}(z_Gqr - y_Gr^2) + \hat{k}(y_Gqr - z_Gq^2)$$

$$\frac{d}{dt}(\hat{j}(rx_G - pz_G)) = \hat{j}(x_G\dot{r} - z_G\dot{p}) + \hat{k}(x_Grp - z_Gp^2) + \hat{i}(z_Grp - x_Gr^2)$$

$$\frac{d}{dt}(\hat{k}(py_G - qx_G)) = \hat{k}(y_G\dot{p} - x_G\dot{q}) + \hat{i}(y_Gpq - x_Gq^2) + \hat{j}(x_Gpq - y_Gp^2)$$

the last two equations were derived from the first following the "right screw" system $x \to y \to z \to x$, $p \to q \to r \to p$. Now taking only the components of $\hat{i}$, adding and equating to X and similarly for $\hat{j}$ and $\hat{k}$:

$$X = m(\dot{u} - vr + wq + z_G\dot{q} - y_G\dot{r} + rpz_G - r^2x_G + pqy_G - q^2x_G)$$

$$X = m(\dot{u} + wq - vr - x_G(r^2+q^2) + y_G(pq-\dot{r}) + z_G(rp+\dot{q}))$$

$$Y = m(\dot{v} + ur - wp - y_G(p^2+r^2) + z_G(qr-\dot{p}) + x_G(pq+\dot{r}))$$

$$Z = m(\dot{w} + vp - uq - z_G(q^2+p^2) + x_G(rp-\dot{q}) + y_G(qr+\dot{p})) \qquad (1\text{-}2)$$

Using: $\vec{M}_G = \frac{d}{dt}\,\overrightarrow{(\text{Ang. Moment})}_G$

the moment $\vec{M}$ at the origin can be expressed as the moment at G plus a moment due to the force F at a radius distance $\vec{R}_G$:

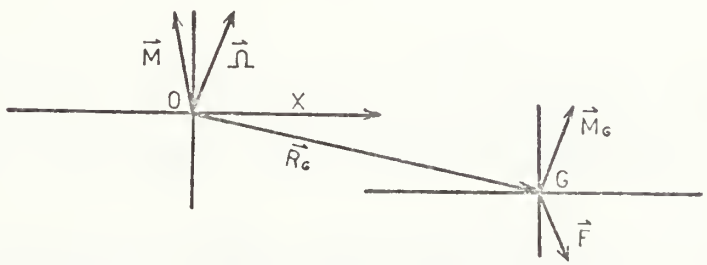


Figure 1-3 Moment Relations

$$\vec{M} = \vec{M}_G + \vec{R}_G \times \vec{F}$$

$$\overrightarrow{(\text{Ang. Moment})}_G = \hat{i}\, I_{xG}p + \hat{j}\, I_{yG}q + \hat{k}\, I_{zG}r$$

the G indicates an origin at the center of gravity

$$I_x = I_{xG} + m(y_G^2 + z_G^2)$$

$$I_y = I_{yG} + m(z_G^2 + x_G^2)$$

$$I_z = I_{zG} + m(x_G^2 + y_G^2)$$

Solving for I_{xG}, I_{yG} and I_{zG}, replacing:

$$\overrightarrow{(\text{Ang. Moment})}_G = -m(\hat{i}(y_G^2+z_G^2)p + \hat{j}(z_G^2+x_G^2)q + \hat{k}(x_G^2+y_G^2)r)$$

$$+ \hat{i}\, I_x p + \hat{j}\, I_y q + \hat{k}\, I_z r$$

$$\vec{M}_G = \vec{M} - \vec{R}_G \times \vec{F} = \frac{d}{dt}\,\overrightarrow{(\text{Ang. Moment})}_G$$

After computations and using the results for derivatives of unit vectors:

$$K = I_x\dot{p} + (I_z-I_y)qr + m[y_G(\dot{w}+pv-qu) - z_G(\dot{v}+ru-pw)]$$

$$M = I_y\dot{q} + (I_x-I_z)rp + m[z_G(\dot{u}+qw-rv) - x_G(\dot{w}+pv-qu)] \quad (1\text{-}3)$$

$$N = I_z\dot{r} + (I_y-I_x)pq + m[x_G(\dot{v}+ru-pw) - y_G(\dot{u}+qw-rv)]$$

the terms:

$(qw-rv)$, $(ru-pw)$, $(pv-qu)$ are centripetal accelerations

$(I_z-I_y)qr$, $(I_x-I_z)rp$, $(I_y-I_x)qp$ are gyroscopic effects.

The relationship for forces and moments can be expressed:

$$\left.\begin{matrix}\text{Forces}\\ \text{Moments}\end{matrix}\right\} = f[\text{Properties of body, properties of motion, properties of fluid}],$$

and for a particular ship, in a given fluid, no excitation forces:

$$\left.\begin{matrix}\vec{F}\\ \vec{M}\end{matrix}\right\} = f(\text{properties of motion})$$

$$= f(x_o,y_o,z_o,\phi,\theta,\psi,u,v,w,p,q,r,\dot{u},\dot{v},\dot{w},\dot{p},\dot{q},\dot{r},\delta,\delta',\delta'')$$

The evaluation of this equation is not easy, thus it is necessary to reduce it into a form that permits its evaluation. This is done by a Taylor expansion, e.g.:

$$f(x) = f(\bar{x}_o) + (x - \bar{x}_o)\frac{df(\bar{x}_o)}{dx} + \frac{(x-\bar{x}_o)^2}{2!}\frac{d^2f(\bar{x}_o)}{dx^2} + \ldots$$

Defining: $D_x^n = \frac{d^n}{dx^n}$ and $(x - \bar{x}_o) = \Delta x$ $\bar{x}_o \equiv$ initial condition

$$f(x) = f(\bar{x}_o) + \Delta x Dx f(\bar{x}_o) + \frac{(\Delta x Dx)^2}{2!} f(\bar{x}_o) + \frac{(\Delta x Dx)^3}{3!} f(\bar{x}_o) + \ldots$$

This is the expression of $e^{\Delta x Dx} f(\bar{x}_o)$

and if, for instance there are three variables:

$$f(x,y,z) = [e^{(\Delta x Dx + \Delta y Dy + \Delta z Dz)}] f(\bar{x}_o, \bar{y}_o, \bar{z}_o)$$

which is expanded:

$$f(x,y,z) = f(\bar{x}_o, \bar{y}_o, \bar{z}_o) + (\Delta x Dx + \Delta y Dy + \Delta z Dz) f(\bar{x}_o, \bar{y}_o, \bar{z}_o)$$

$$+ \frac{(\Delta x Dx + \Delta y Dy + \Delta z Dz)^2}{2!} f(\bar{x}_o, \bar{y}_o, \bar{z}_o) \cdots$$

For the case of f(properties of motion) it is linearized and all 2nd and higher order terms are neglected. This amounts to say, that Δx, Δy, Δz, ... etc., are small and restricts the variation around an initial equilibrium condition.

Taking the component of the force along the y axis as an example:

$$X = f(x_o, y_o, z_o, \phi, \theta, \ldots, \dot{v}, \ldots, \delta'')$$

the linear terms are obtained:

$$X = f(\ldots)_o + \Delta x_o \left(\frac{\partial f}{\partial x_o}\right)_o + \Delta y_o \left(\frac{\partial f}{\partial y_o}\right)_o + \ldots + \Delta v \left(\frac{\partial f}{\partial v}\right)_o + \ldots$$

defining:

$$\left(\frac{\partial f}{\partial u}\right)_o = \left(\frac{\partial X}{\partial u}\right)_{u=u_o} = X_u$$

$$\vdots$$

$$\left(\frac{\partial X}{\partial \omega}\right)_{\omega=\omega_o=0} = X_\omega$$

$$\vdots$$

$$\Delta\omega = (\omega - \omega_o) = \omega \quad \text{All initial conditions zero except } u_o$$

$$\vdots$$

$$\Delta u = (u - u_o) = \Delta u$$

In similar fashion X,Z (Force components along x and z axis) can be derived and the Force equations become:

$$X = X_o + X_{x_o}x_o + X_{y_o}y_o \ldots X_\theta\theta + \ldots X_u\Delta u \ldots$$

$$Y = Y_o + Y_{x_o}x_o + Y_{y_o}y_o \ldots Y_\theta\theta + \ldots Y_u\Delta u \ldots \qquad (1\text{-}4)$$

$$Z = Z_o + Z_{x_o}x_o + Z_{y_o}y_o \ldots Z_\theta\theta + \ldots Z_u\Delta u \ldots$$

A similar derivation can be done for K,M,N (Moment components).

It is now necessary to equate Eqs. (1-2) and (1-4), but Eq. (1-2) must be linearized first. The equations can be further simplified assuming the center of gravity at $x_G=0$, $y_G=0$, $z_G=0$.

The process of linearization is shown next for the case of the Y force and motion without rolling or pitch.

$$Y = m(\dot{v} + ur - wp + x_G(pq+\dot{r}))$$

$$Y = m(\dot{v} + ur + x_G\dot{r})$$

$$\dot{v} + ru + x_G\dot{r} = (\dot{v}_o+\Delta\dot{v}) + (r_o+\Delta r)(u_o+\Delta u) + x_G(\dot{r}_o+\Delta\dot{r})$$

$$= \Delta\dot{v} + \Delta r u_o + \Delta r\Delta u + x_G\Delta\dot{r}$$

Letting

$$\dot{v}_o = \dot{r}_o = r_o = 0,\ \Delta\dot{v} = \dot{v},\ \Delta r = r,\ \Delta\dot{r} = \dot{r}$$

$\Delta r\Delta u$ is dropped (non linear)

$$Y = m(\dot{v} + ru_o + x_G\dot{r})$$

Now from the function equation Eq. (1-4) and also for the condition of no rolling (ϕ=0), no pitch (θ=0):

$$Y_{x_o}x_o+Y_{y_o}y_o+Y_\psi\psi+Y_u u+Y_v v+Y_r r+Y_{\dot{u}}\dot{u}+Y_{\dot{v}}\dot{v}+Y_{\dot{r}}\dot{r} = m(\dot{v}+ru_o+x_G\dot{r}) \quad (1\text{-}5)$$

Similar considerations can be applied to the other components.

B. EXAMPLE

As a specific example, the "Twin Hulled Semisubmerged ship" [Ref. 4] is considered next.

A summary of the coordinate system and equations of motions is reproduced from Ref. 4:

1. The ship fixed axes have their origin at the center of mass.
2. Force and moment equations are written in the ship fixed axis system.
3. The ship is assumed to be running at a constant velocity U=u and all other motion variables (velocities, angular

rates, attitude angles) are assumed small except the heading angle ψ.

4. The equations of motion are written in a nondimensional form based on Froude number scaling, the nondimensional variables being designated by a $\hat{}$.

5. The small perturbation assumption allows the equations of motion to be linearized and the equations to be separated into two groups: the lateral-directional (roll-yaw-sway) equations and the longitudinal (pitch-heave) equations.

In order to obtain the equations in a non dimensional form some definitions will be given, and applied to the Y force equation Eq. (1-5) as an example of the process.

$$\text{Froude number} = \frac{U}{\sqrt{g\ell}} \qquad \ell = \text{characteristic length}$$

$$\hat{u},\hat{v},\hat{w} = \frac{u,v,w}{\sqrt{g\ell}}$$

$$\hat{p},\hat{q},\hat{r} = \frac{p,q,r}{\sqrt{g/\ell}}$$

$$\tau = t\sqrt{g/\ell}$$

$$C_X, C_Y, C_Z = \frac{X, Y, Z}{\frac{1}{2}\rho U^2 \ell^2}$$

$$C_{x\square}, C_{y\square}, C_{z\square} = \frac{\partial}{\partial \square}(C_X, C_Y, C_Z)$$

$$\hat{X},\hat{Y},\hat{Z} = \frac{X,Y,Z}{\rho g \ell^3}$$

$$\hat{X}_\square, \hat{Y}_\square, \hat{Z}_\square = \frac{\partial}{\partial \square}(\hat{X},\hat{Y},\hat{Z})$$

The equation for the side force Y is written now allowing variation in rolling (known to be small) and rudder deflection (δ_r). Also $x_G=0$ for C G as origin of system.

$$m(\dot{v}+ru) = (Y_v v+Y_p p+Y_r r+Y_{\delta_r}\delta_r) + (Y_{\dot{v}}\dot{v}+Y_{\dot{p}}\dot{p}+Y_{\dot{r}}\dot{r}) \qquad (1\text{-}6)$$

Note that $Y_\psi \psi$ does not appear in the present equation because the orientation does not affect the lateral dynamics.

Note also that this is the same equation that was derived before with the new considerations. Using the above relations:

$$Y_v = \frac{dY}{dv}$$

$$C_y = \frac{Y}{\frac{1}{2}\rho U^2 \ell^2}$$

$$\hat{v} = \frac{v}{\sqrt{g\ell}}$$

$$Y_v = \frac{1}{2}\frac{\rho \ell^2 U^2}{\sqrt{g\ell}} \frac{\partial\left(\frac{Y}{\frac{1}{2}\rho U^2 \ell^2}\right)}{\partial\left(\frac{v}{\sqrt{g\ell}}\right)} = \frac{1}{2}\frac{\rho \ell^2 U^2}{\sqrt{g\ell}} C_{y\hat{v}}$$

$$Y_{\delta_r} = \frac{\partial Y}{\partial(\delta_r)}$$

$$= \frac{1}{2}\rho U^2 \ell^2 \frac{\partial\left(\frac{Y}{\frac{1}{2}\rho U^2 \ell^2}\right)}{\partial(\delta_r)}$$

$$= \frac{1}{2}\rho U^2 \ell^2 C_{y\delta r}$$

Similarly:

$$Y_r = \frac{dY}{dr}$$

$$\hat{r} = \frac{r}{\sqrt{g/\ell}} = \frac{r\ell}{\sqrt{g\ell}}$$

$$Y_r = \tfrac{1}{2}\rho \frac{\ell^3 U^2}{\sqrt{g\ell}} \frac{\partial\left(\frac{Y}{\frac{1}{2}\rho U^2 \ell^2}\right)}{\partial\left(\frac{r\ell}{\sqrt{g\ell}}\right)} = \tfrac{1}{2} \frac{\rho U^2 \ell^3}{\sqrt{g\ell}} C_{Y_r}$$

$$r = \frac{\sqrt{g\ell}}{\ell} \hat{r} \qquad \text{similar result for p}$$

$$Y_{\dot{r}} = \frac{\partial Y}{\partial \dot{r}}$$

$$\hat{\dot{r}} = \frac{\dot{r}}{g/\ell} = \frac{\dot{r}\ell^2}{g\ell}$$

$$\frac{\partial Y}{\partial \dot{r}} = \frac{\partial\left(\frac{Y}{\rho g \ell^3}\right)}{\partial\left(\frac{\dot{r}}{\rho g \ell^3}\right)} = \rho\ell^4 \frac{\partial \hat{Y}}{\partial\left(\frac{\dot{r}\ell}{g}\right)} = \rho\ell^4 \hat{Y}_{\dot{r}}$$

$$\tau = t\sqrt{g/\ell}$$

$$\frac{d\tau}{dt} = \sqrt{g/\ell}$$

$$\frac{d\hat{r}}{d\tau} = \frac{1}{\sqrt{g/\ell}} \frac{dr}{dt} \frac{dt}{d\tau} \qquad \text{or:}$$

$$\hat{\dot{r}} = \frac{\dot{g}r}{\ell} \qquad \text{A similar result is obtained for } \dot{p}.$$

$$Y_{\dot{v}} = \frac{\partial Y}{\partial \dot{v}} = \frac{\partial \left(\frac{Y}{\rho g \ell^3}\right)}{\partial \left(\frac{\dot{v}}{\rho g \ell^3}\right)} = \rho \ell^3 \frac{\partial \hat{Y}}{\partial \left(\frac{\dot{v}}{g}\right)}$$

$$Y_{\dot{v}} = \rho \ell^3 \hat{Y}_{\hat{\dot{v}}}$$

$$\dot{v} = g\hat{\dot{v}}$$

Replacing in the left side of Eq. (1-6)

$$m(\dot{v} + ru) = m\left(g\hat{\dot{v}} + \frac{\sqrt{g\ell}}{\ell} \hat{r} \sqrt{g\ell}\ \hat{U}\right)$$

For the right side of Eq. (1-6) the following expressions are obtained:

$$Y_v v = \tfrac{1}{2}\rho \ell^2 U^2 C_{Y_{\hat{v}}} \hat{v} \qquad Y_p p = \tfrac{1}{2}\rho U^2 \ell^2 C_{Y_{\hat{p}}} \hat{p}$$

$$Y_{\dot{v}} \dot{v} = \rho g \ell^3 Y_{\hat{\dot{v}}} \hat{\dot{v}} \qquad Y_{\dot{p}} \dot{p} = \rho g \ell^3 \hat{Y}_{\hat{\dot{p}}} \hat{\dot{p}}$$

$$Y_r r = \tfrac{1}{2}\rho U^2 \ell^2 C_{Y_{\hat{r}}} \hat{r} \qquad Y_{\dot{r}} \dot{r} = \rho g \ell^3 \hat{Y}_{\hat{\dot{r}}} \hat{\dot{r}}$$

$$m = \rho \ell^3$$

After replacing and adding the effect of waves ($\hat{Y}_w$), Eq. (1-6) becomes:

$$\hat{\dot{v}} + \hat{r}\hat{U} = \tfrac{1}{2}\hat{U}^2 (C_{Y_{\hat{v}}} \hat{v} + C_{Y_{\hat{p}}} \hat{p} + C_{Y_{\hat{r}}} \hat{r} + C_{Y_{\delta_r}} \delta r) + (\hat{Y}_{\hat{\dot{p}}} \hat{\dot{p}} + \hat{Y}_{\hat{\dot{v}}} \hat{\dot{v}} + \hat{Y}_{\hat{\dot{r}}} \hat{\dot{r}} + \hat{Y}_w) \qquad (1\text{-}7)$$

Values for the coefficients can be obtained from the report [Ref. 4]. The other equations for rolling Moment,

Yawing Moment (which with the side force are the lateral Directional Dynamics) and for Longitudinal Dynamics: Vertical force, pitching moment are derived in a similar way to the equation just derived here.

Now it is interesting to derive the transfer function from the above equations.

The procedure is illustrated considering the longitudinal dynamics:

vertical force:

$$\dot{\hat{w}} - \hat{U}\hat{q} = \tfrac{1}{2}\hat{U}^2(C_{z\hat{w}}\hat{w}+C_{z\hat{q}}\hat{q}+C_{z\delta e}\delta e+C_{z\delta c}\delta c) - \hat{A}_{wL}\hat{z}+\hat{A}_w\bar{\hat{X}}_{wL}\theta+Z_{\dot{\hat{w}}}\dot{\hat{w}} + Z_{\dot{\hat{q}}}\dot{\hat{q}} + \hat{Z}_w$$

pitching moment:

$$\hat{P}_y^2\dot{\hat{q}} = \tfrac{1}{2}\hat{U}^2(C_{m\hat{w}}\hat{w}+C_{m\hat{q}}\hat{q}+C_{m\delta e}\delta e+C_{m\delta c}\delta c) - Z_{MP}\theta+\hat{A}_{wL}\bar{\hat{X}}_{wL}\hat{z}+M_{\dot{\hat{w}}}\dot{\hat{w}} + M_{\dot{\hat{q}}}\dot{\hat{q}} + \hat{M}_w$$

using $\dot{\theta} = \hat{q}$ and $\hat{w} = \dot{\hat{z}} + \hat{U}\theta$:

$$\hat{z}[-\hat{A}_{wL}+\tfrac{1}{2}\hat{U}^2C_{z\hat{w}}S+(Z_{\dot{\hat{w}}}-1)S^2] + \theta[\hat{A}_w\bar{\hat{X}}_{wL}+\tfrac{1}{2}\hat{U}^3C_{z\hat{w}} + (\tfrac{1}{2}\hat{U}^2C_{z\hat{q}}+\hat{U}Z_{\dot{\hat{w}}})S+Z_{\dot{\hat{q}}}S^2] = -\tfrac{1}{2}\hat{U}^2[C_{z\delta e}\delta e+C_{z\delta c}\delta c] - \hat{Z}_w$$

$$\hat{z}[\hat{A}_{wL}\bar{\hat{X}}_{wL}+\tfrac{1}{2}\hat{U}^2C_{m\hat{w}}S+M_{\dot{\hat{w}}}S^2] + \theta[-Z_{MP}+\tfrac{1}{2}\hat{U}^3C_{m\hat{w}} + (\tfrac{1}{2}\hat{U}^2C_{m\hat{q}}+\hat{U}M_{\dot{\hat{w}}})S+(M_{\dot{\hat{q}}}-\hat{P}_y^2)S^2] = -\tfrac{1}{2}\hat{U}^2[C_{m\delta e}\delta e+C_{m\delta c}\delta c] - \hat{M}_w$$

(1-8)

The equations (1-8) can be written in general (only one forcing function):

$$\hat{z}[L_1+L_2S+L_3S^2] + \theta[L_4+L_5S+L_6S^2] = I_1$$

$$\hat{z}[L_7+L_8S+L_9S^2] + \theta[L_{10}+L_{11}S+L_{12}S^2] = I_1$$

$$\theta = \frac{\begin{vmatrix} I_1 & L_4+L_5S+L_6S^2 \\ I_1 & L_{10}+L_{11}S+L_{12}S^2 \end{vmatrix}}{\begin{vmatrix} L_1+L_2S+L_3S^2 & L_4+L_5S+L_6S^2 \\ L_7+L_8S+L_9S^2 & L_{10}+L_{11}S+L_{12}S^2 \end{vmatrix}}$$

$$\theta = \frac{I_1((L_{12}-L_6)S^2 + (L_{11}-L_5)S + (L_{10}-L_4))}{\Delta}$$

finally after solving for Δ and rearranging:

$$\frac{\theta}{I_1} = \frac{C_3S^2 + C_2S + C_1}{S^4 + C_4S^3 + C_5S^2 + C_6S + C_7} \qquad (1\text{-}9)$$

The values of C_1, C_2, C_3, C_4, C_5, C_6, C_7 can be obtained using the computer program developed in FORMAC [Ref. 8].

II. MODELING TECHNIQUE

A. DEVELOPMENT AND STANDARDIZATION

Having an equation of the form:

$$(\alpha S^n + \beta S^{n-1} + \gamma S^{n-2} \ldots \zeta)a = I \qquad (2\text{-}1a)$$

In order to simulate the equation in an analog or digital computer, $\frac{d^n a}{dt^n}$ is obtained:

$$\frac{d^n a}{dt^n} = -\frac{1}{\alpha}\left(\beta \frac{d^{n-1}a}{dt^{n-1}} + \gamma \frac{d^{n-2}a}{dt^{n-2}} + \ldots \zeta a\right) + \frac{I}{\alpha} \qquad (2\text{-}1b)$$

From Eq. (2-1b) the block diagram is obtained:

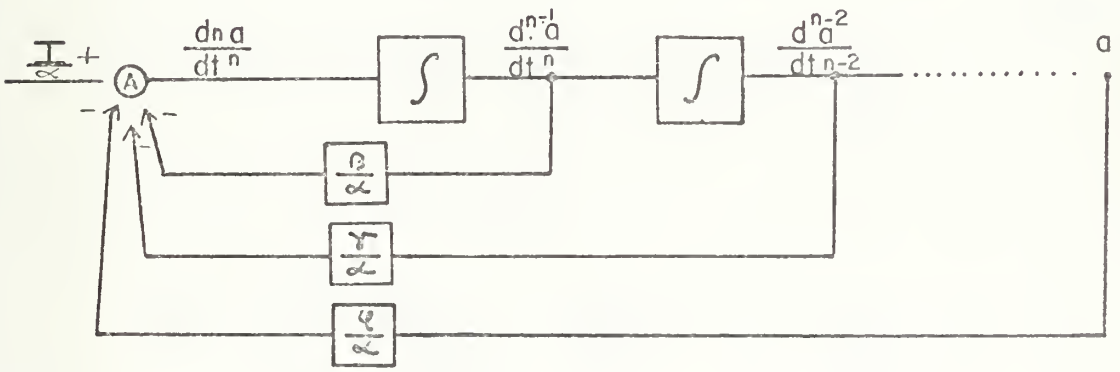


Figure 2-1 Block diagram Eq. (2-1b)

All the inputs to node A are divided by α so the above diagram can be converted to:

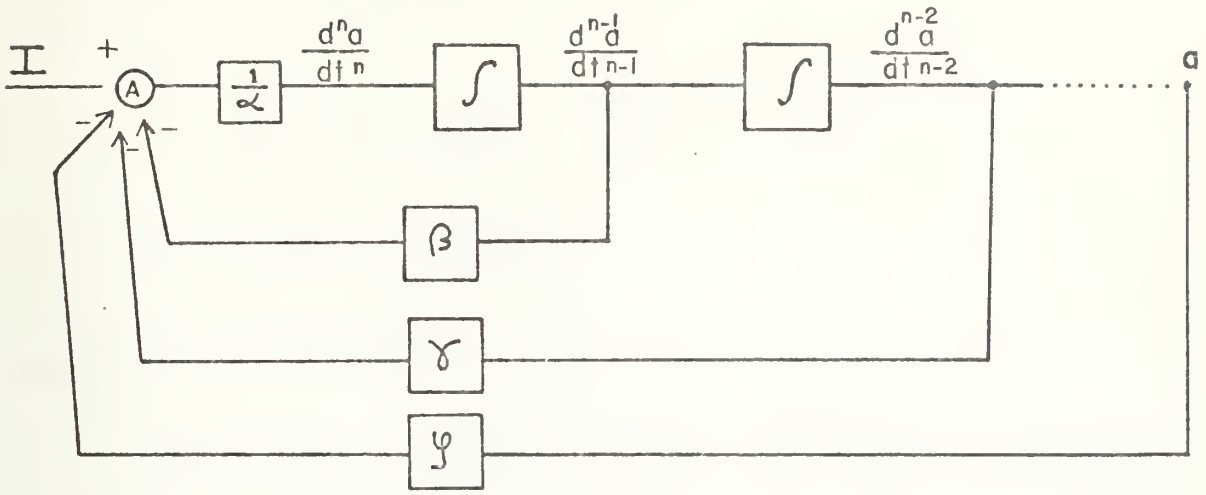


Figure 2-2 Block diagram Eq. (2-1b)

Equation (2-1a) could also be rearranged:

$$aS^n = -\frac{1}{\alpha}\left(\beta S^{n-1} + \gamma S^{n-2} \ldots \zeta\right)a + \frac{I}{\alpha} \qquad (2\text{-}2)$$

$$a = -\frac{1}{\alpha}\left(\frac{\beta}{S} + \frac{\gamma}{S^2} \cdots \frac{\zeta}{S^n}\right)a + \frac{I}{\alpha S^n}$$

whose block diagram is simply:

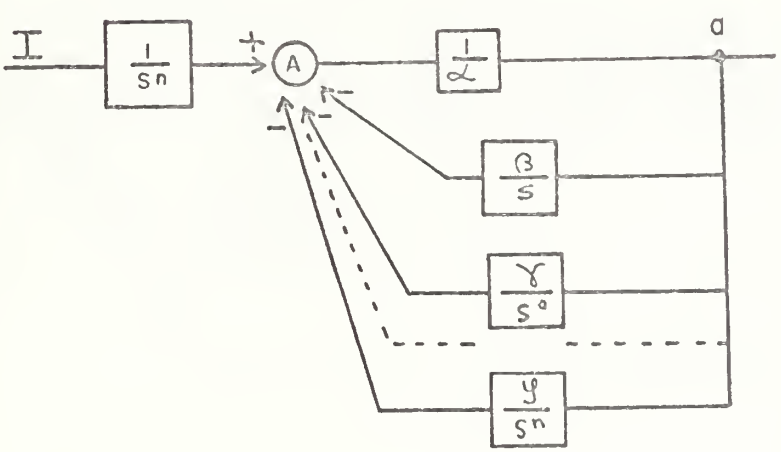


Figure 2-3 Block diagram Eq. (2-2)

Equation (2-2) is of the form:

$$a\left(1 + \frac{1}{\alpha}\left(\frac{\beta}{S} + \frac{\gamma}{S^2} \cdots \frac{\zeta}{S^n}\right)\right) = \frac{I}{\alpha S^n}$$

or $a(1 + G(S)) = I(S)$ (2-3)

For a set of equations:

$$\begin{bmatrix} (1 + G_{aA}) & + & G_{bA} \\ G_{aB} & + & (1 + G_{bB}) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} K_a I_n \\ K_b I_n \end{bmatrix} \quad (2\text{-}4)$$

Solving for a in the first row and for b in the second:

$$a = K_a I_n - G_{aA} a - G_{bA} b$$

$$b = K_b I_n - G_{bB} b - G_{aB} a \quad (2\text{-}5a)$$

The block diagram for Eq. (2-5a):

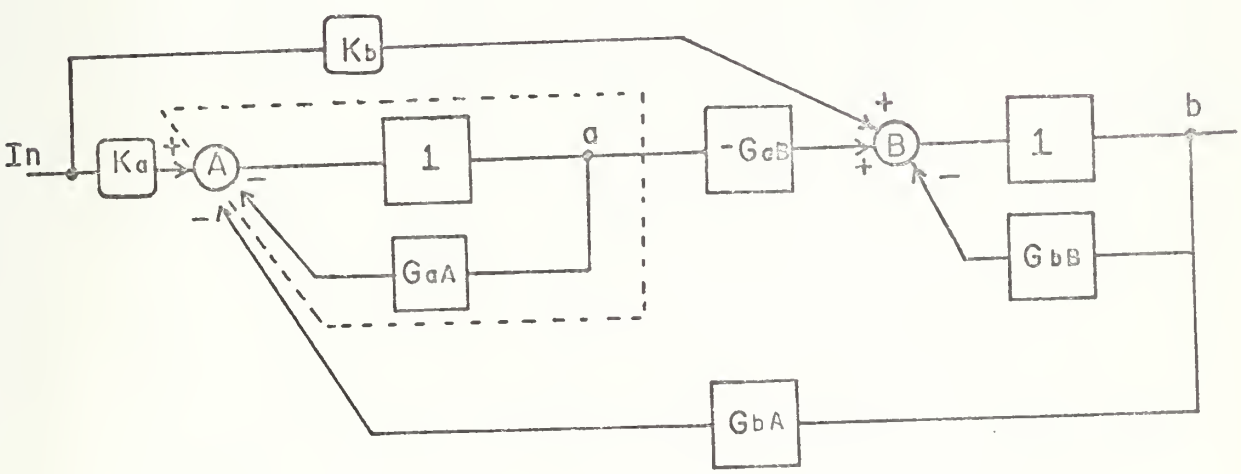


Figure 2-4a Block diagram Eq. (2-5a)

It can be noted from Eq. 2-4 the relation of the terms in the main diagonal, above and below the diagonal with respect to the block diagram. Then for a system of n equations any of the rows can be written:

$$b = K_n I_n - G_{bB}b - G_{aB}a - G_{cB}c \ldots G_{nB}n \qquad (2\text{-}5b)$$

which is drawn

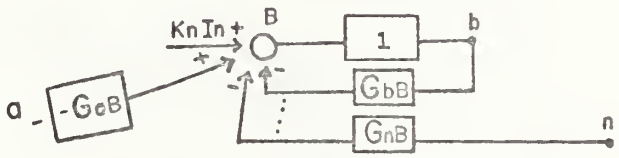


Figure 2-4b Block diagram Eq. (2-5b)

This agrees with the equations in the report No. 20 by Dr. G. J. Thaler [Ref. 10].

The above results can be applied to:

$$\begin{bmatrix} 1+G_{aA} & G_{bA} & G_{cA} \ldots\ldots & G_{nA} \\ G_{aB} & 1+G_{bB} & G_{cB} \ldots\ldots & G_{nB} \\ G_{aC} & G_{bC} & 1+G_{cC} \ldots\ldots & G_{nC} \\ \vdots & \vdots & \vdots & \vdots \\ G_{aN} & G_{bN} & G_{cN} \ldots\ldots & 1+G_{nN} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ \vdots \\ n \end{bmatrix} = \begin{bmatrix} K_a \\ K_b \\ K_c \\ \vdots \\ K_n \end{bmatrix} \left[I_1 + \ldots I_n \right] \qquad (2\text{-}5c)$$

In which G's and K's are functions of S and a, b, ..., n are the variables of interest and in a physical system they can represent: velocity, position, etc. I's are forcing functions. The terms in the main diagonal are drawn as

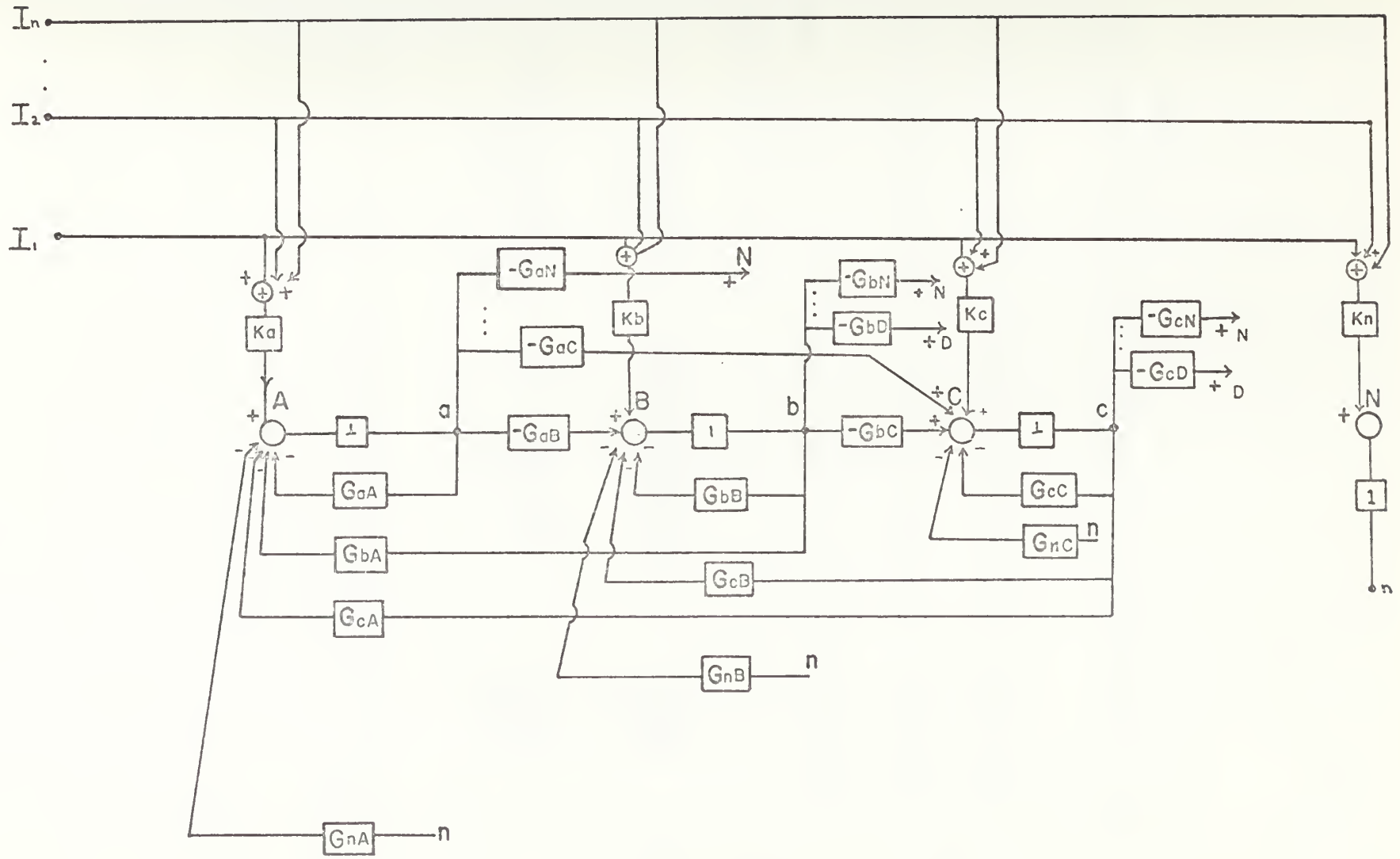


Figure 2-5 Block diagram Eq. (2-5c)

shown in the dashed portion of Fig. 2-4a, the terms above the diagonal are feedback terms, and below the diagonal are forward terms as indicated by the subscripts. Figure 2-5 shows the diagram of a set of n equations.

Normally the terms in the matrix are not found in the form shown but instead are given in a polynomial form, e.g.:

$$\begin{aligned}(\gamma_{aA}+\beta_{aA}S+\alpha_{aA}S^2)a + (\gamma_{bA}+\beta_{bA}S+\alpha_{bA}S^2)b &= I_1 + I_2 \\ (\gamma_{aB}+\beta_{aB}S+\alpha_{aB}S^2)a + (\gamma_{bB}+\beta_{bB}S+\alpha_{bB}S^2)b &= I_1 + I_2\end{aligned} \qquad (2\text{-}6a)$$

But this set of equations can be converted into the desired form:

$$\begin{aligned}\left(1 + \frac{\gamma_{aA}+\beta_{aA}S}{\alpha_{aA}S^2}\right)a + \left(\frac{\gamma_{bA}+\beta_{bA}S+\alpha_{bA}S^2}{\alpha_{aA}S^2}\right)b &= \frac{I_1+I_2}{\alpha_{aA}S^2} \qquad (2\text{-}6b) \\ \left(\frac{\gamma_{aB}+\beta_{aB}S+\alpha_{aB}S^2}{\alpha_{bB}S^2}\right)a + \left(1 + \frac{\gamma_{bB}+\beta_{bB}S}{\alpha_{bB}S^2}\right)b &= \frac{I_1+I_2}{\alpha_{bB}S^2}\end{aligned}$$

From the previous results the block diagram is drawn very simply as shown in Figure 2-6. Note that all inputs and feedbacks to any summer I are divided by $\alpha_{iI}S^2$.

A general representation of any summer-node is given in Figure 2-7. This is due to the fact that all the inputs to a summer (the row of the matrix), were divided by $\alpha_{iI}S^2$, which corresponds to the highest order term in the main

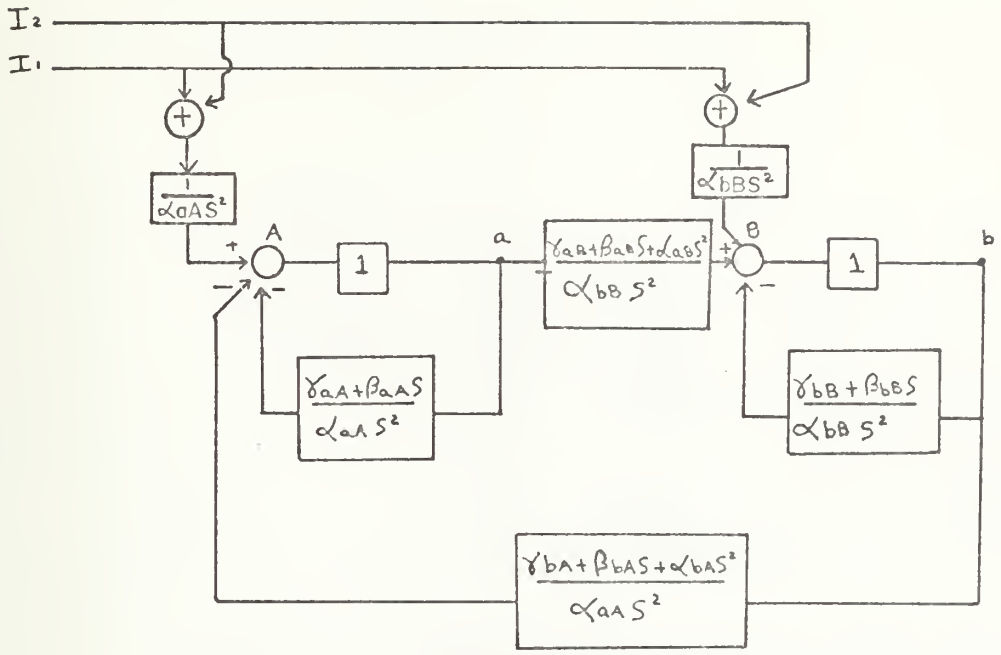


Figure 2-6 Block diagram Eq. (2-6b)

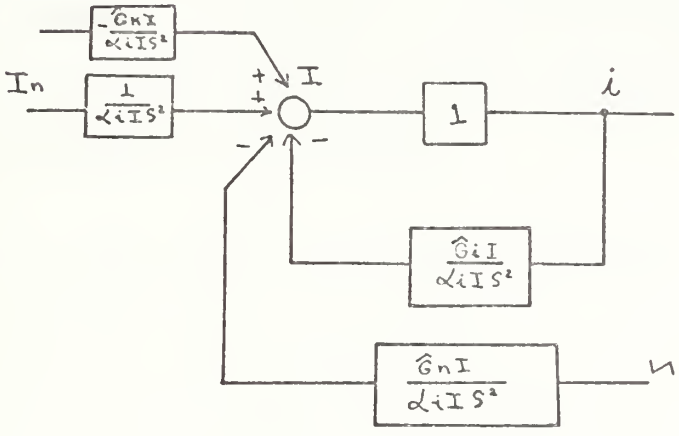


Figure 2-7 Summer-Node Representation

diagonal, in order to convert the equations into the desired forms (see Eqs. (2-6a) and (2-6b)). Accordingly this last block can be converted into:

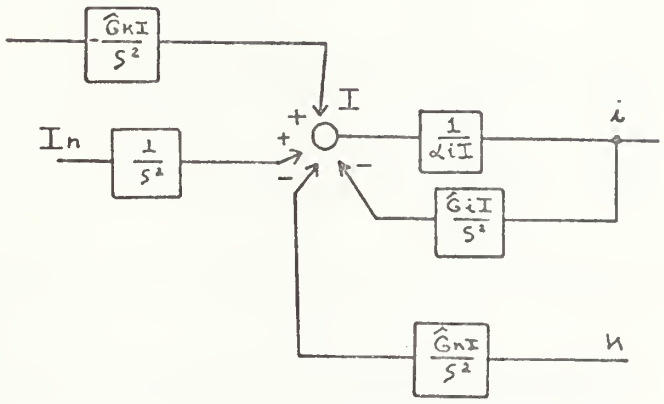


Figure 2-8 Summer-Node Representation (2)

This, then implies that the block diagram can be obtained directly from the set of equations in the form of Eq. (2-6a) and for that the rules are as follows:

Polynomials of the form: $\alpha_{iI}S^n+\beta_{iI}S^{n-1}+\gamma_{iI}S^{n-2}+\ldots\zeta$

1. Draw a set of summers and nodes depending on the number of equations (or variables)

   A a B b ... N n

2. In each row, place the polynomial containing the highest order in S, in the diagonal.
3. Terms in the diagonal of the matrix are shown in Figure 2-9. Note that the block in the forward path is the inverse of the coefficient of the highest order term of the polynomial in the main diagonal corresponding to row I (see Eq. (2-6)). S^n is the highest order term.

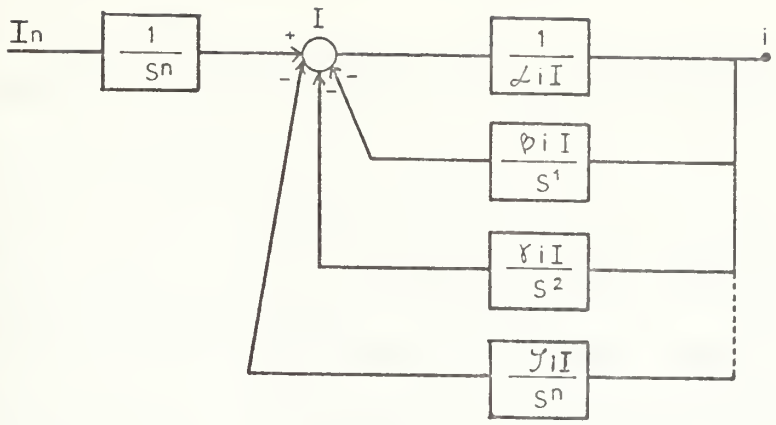


Figure 2-9 Block, Terms in Diagonal

4. Terms above diagonal (Feedback terms):

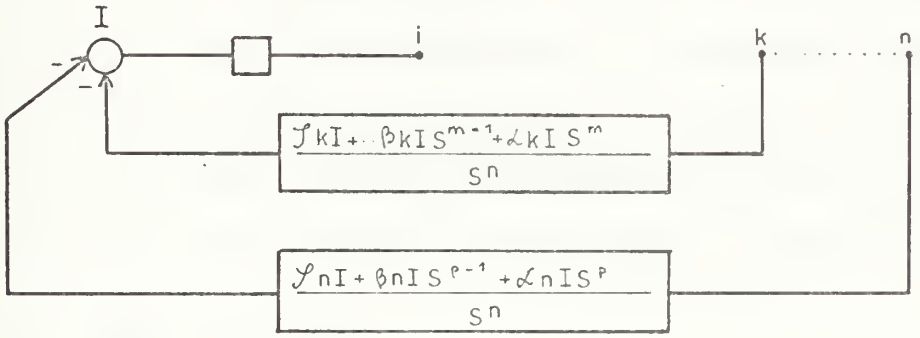


Figure 2-10 Block, Terms above Diagonal

5. Terms below diagonal (Forward terms):

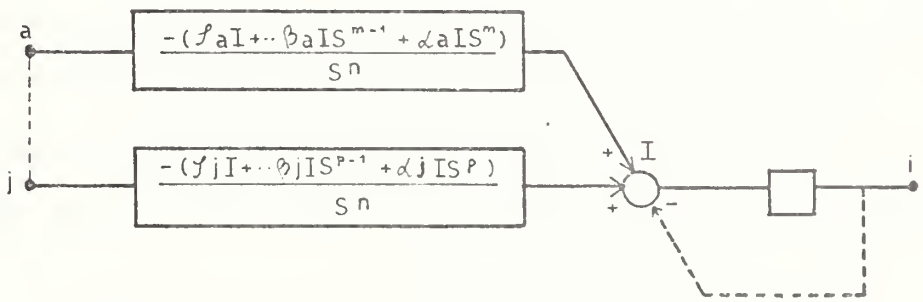


Figure 2-11 Block, Terms below Diagonal

Note: The subscripts indicate the direction of the path, e.g., γ_{aI} $a \to I$.

Application to a system of three equations and 2nd order polynomials:

$$\begin{aligned}
(\alpha_{aA}S^2+\beta_{aA}S+\gamma_{aA})a + (\alpha_{bA}S^2+\beta_{bA}S+\gamma_{bA})b + (\alpha_{cA}S^2+\beta_{cA}S+\gamma_{cA})c &= I_1 + I_2 \\
(\alpha_{aB}S^2+\beta_{aB}S+\gamma_{aB})a + (\alpha_{bB}S^2+\beta_{bB}S+\gamma_{bB})b + (\alpha_{cB}S^2+\beta_{cB}S+\gamma_{cB})c &= I_2 \qquad (2\text{-}7) \\
(\alpha_{aC}S^2+\beta_{aC}S+\gamma_{aC})a + (\alpha_{bC}S^2+\beta_{bC}S+\gamma_{bC})b + (\alpha_{cC}S^2+\beta_{cC}S+\gamma_{cC})c &= I_1
\end{aligned}$$

The block diagram is shown in Fig. 2-12. The dotted lines will be discussed later. This block diagram is suitable for simulation in analog computer since all the components are integrators or potentiometers, no derivatives appear. The best application for simulation is using a digital computer with CSMP (Continuous System Modeling Program), although some restrictions will be given in Section III.

B. NON LINEAR EQUATIONS

Application to a nonlinear equation:

$$\begin{aligned}
\alpha_{aA}S^2a + \alpha_{bA}Sb &= I - \zeta ab \\
\alpha_{aB}\ a + \alpha_{bB}Sb &= \sigma a^2
\end{aligned} \qquad (2\text{-}8)$$

Note that the inputs are function of the variables a and b, and the left side has the variables in linear form. Following the rules the block diagram is shown in Fig. 2-13. This diagram can be implemented very easily in an analog or

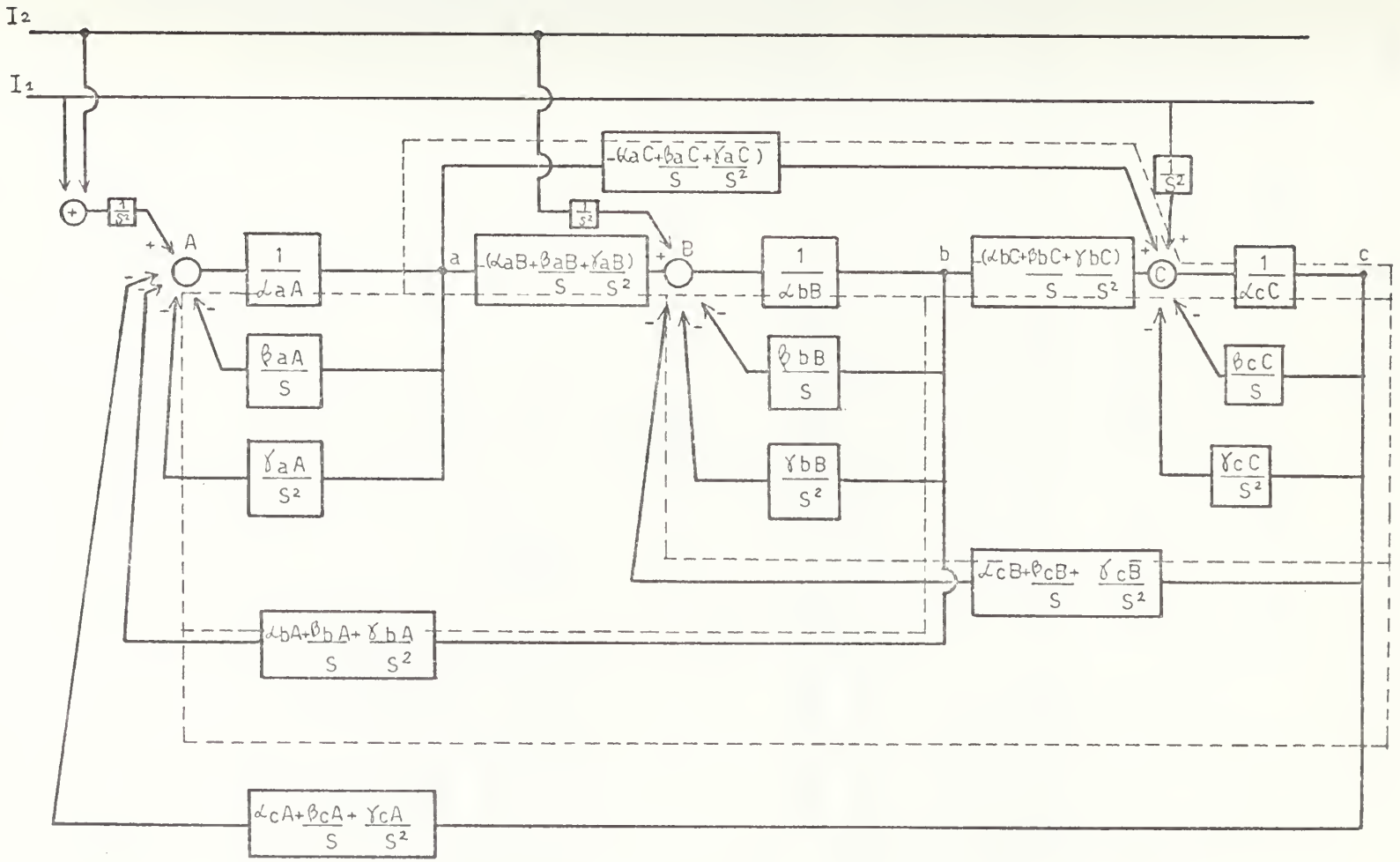


Figure 2-12 Block diagram Eq. (2-7)

in a digital computer. An example of a simulation using CSMP is given [Ref. 5]:

$$\begin{aligned} (-x + \frac{dx}{dt}) - y &= -x^3 - xy^2 \\ x + (-y + \frac{dy}{dt}) &= -y^3 - yx^2 \end{aligned} \qquad (2\text{-}9)$$

The computer plottings of x vs. y are shown in Fig. 2-15 for two different initial conditions, and the block diagram is Fig. 2-14.

The results of Fig. 2-15 are of course the expected solution since transforming Eq. (2-9) into polar coordinates:

$$\frac{dr}{dt} = -r(r^2-1) \quad , \quad \frac{d\theta}{dt} = -1$$

It is clear that when $r>1$, $\frac{dr}{dt}$ is negative and when $r<1$, $\frac{dr}{dt}$ is positive; this indicates that the forces acting on the system are such that r will decrease when $r>1$, and increase r when $r<1$. As soon as $r=1$, $\frac{dr}{dt} = 0$ and a limit cycle (stable) is reached; this situation is described by the equation $r=1$, $\theta = \theta_o + t$ and can last indefinitely.

C. EXTENSIONS

Sometimes the derivatives of the variables must be available which must be done without the use of differentiators. This can be done by a change in variables.

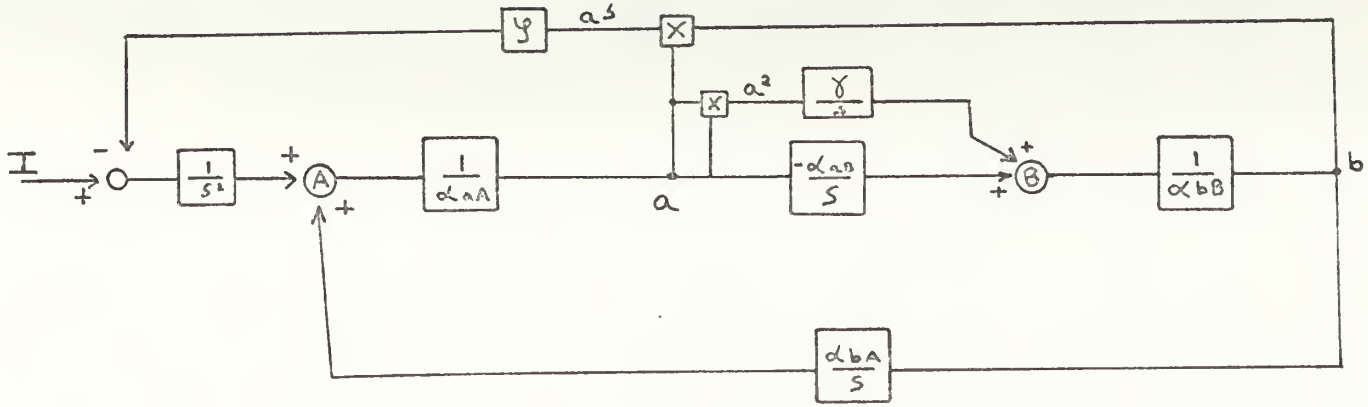


Figure 2-13 Block diagram Eq. (2-8)

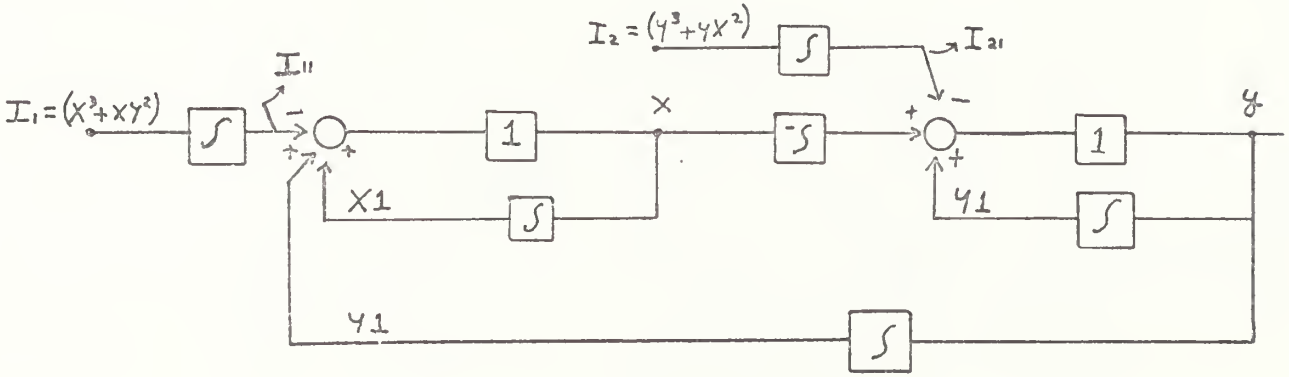


Figure 2-14 Block diagram Eq. (2-9)

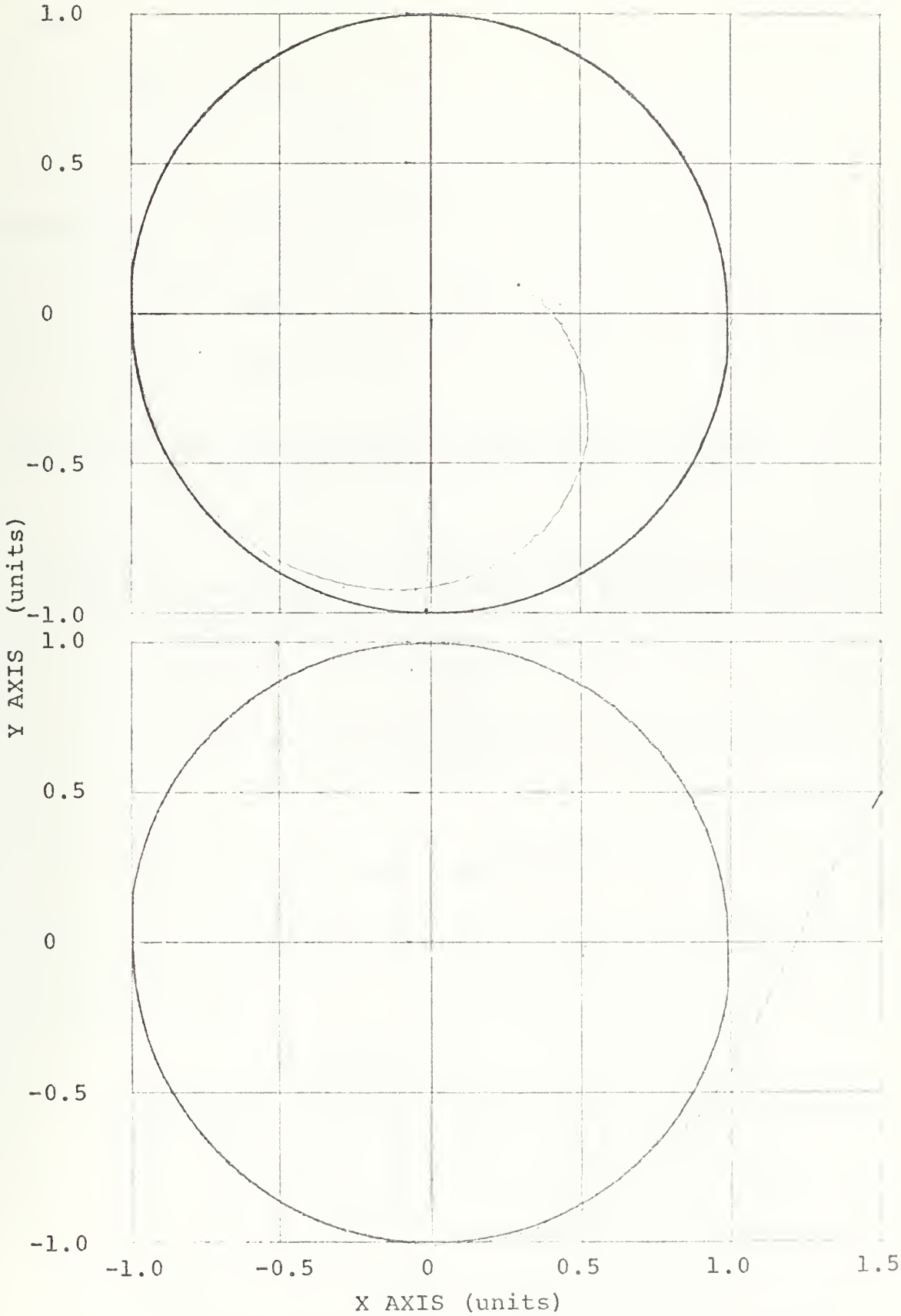


Figure 2-15 Computer solution of Eq. (2-9)

The procedure is best illustrated with an example:

Given:

$$\begin{aligned} \alpha_{aA}S^2a + (\alpha_{bA}S^2 + \beta_{bA})b &= I \\ \alpha_{aB}Sa + \alpha_{bB}S^2b &= 0 \end{aligned} \qquad (2\text{-}10)$$

These equations can be written:

$$\begin{aligned} \alpha_{aA}S\dot{a} + \alpha_{bA}\ddot{b} &= I - \beta_{bA}b \\ \alpha_{aB}\dot{a} + \alpha_{bB}\ddot{b} &= 0 \end{aligned} \qquad (2\text{-}11)$$

From here the block diagram is drawn in Fig. 2.16a.

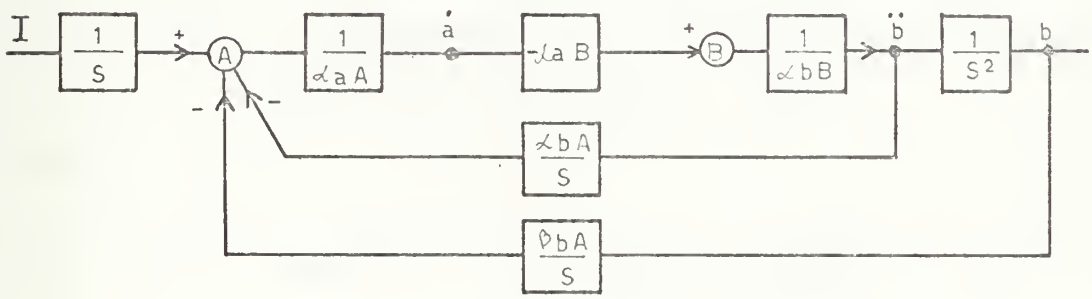


Figure 2-16a Block diagram Eq. (2-11)

As a proof $\frac{b}{I}$ can be obtained by block transformation of Fig. 2-16a:

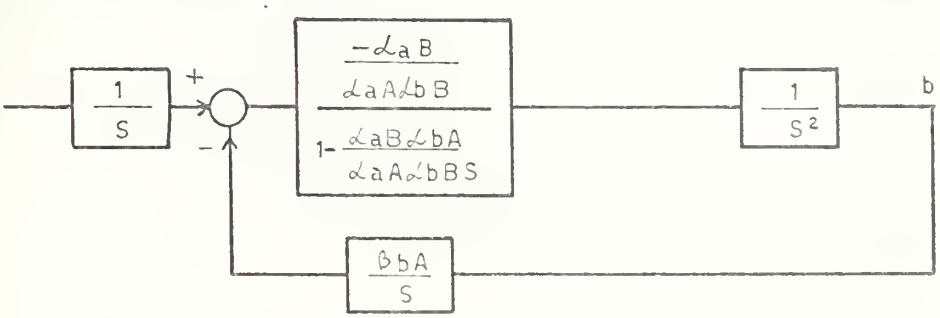


Figure 2-16b Block transformation of Fig. 2-16a

$$\frac{b}{I} = \frac{-\alpha_{aB}}{\alpha_{aA}\alpha_{bB}S^3 - \alpha_{aB}\alpha_{bA}S^2 - \alpha_{aB}\beta_{bA}}$$

Same result is obtained solving Eq. (2-10) by Cramer's Rule.

In summary, if some derivatives are wanted, it is possible to rearrange the equations with new parameters, which are the highest derivatives and the remaining lower derivatives and the unknown itself are considered as inputs. In doing this, it is necessary to remember that the highest order polynomial must be in the diagonal (always), otherwise the use of differentiators cannot be avoided. This implies that all the rows of the matrix have to be checked in order to fulfill the requirement, e.g.:

Given:

$$(\gamma_{aA} + \beta_{aA}S + \alpha_{aA}S^2)a + (\gamma_{bA} + \beta_{bA}S + \alpha_{bA}S^2)b = I_1 \qquad (2\text{-}12)$$

$$(\gamma_{aB} + \beta_{aB}S + \alpha_{aB}S^2)a + (\alpha_{bB}S)b = 0$$

The block diagram from Eq. (2-12):

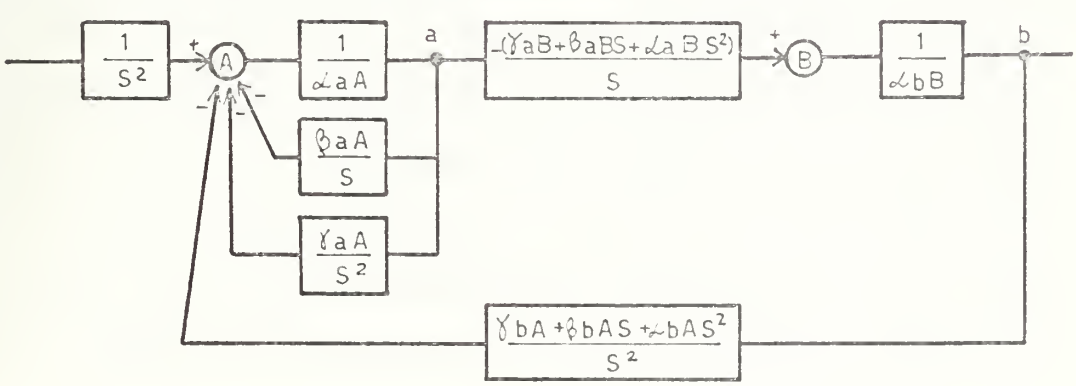


Figure 2-17 Block diagram Eq. (2-12)

It is seen that the block between a and B has a differentiator (the highest order polynomial in the second row was not in the main diagonal).

An interchange between the rows or the columns would give the desired configuration to fulfill the requirement.

From Eq. (2-12) as it is, derivatives cannot be obtained without the use of differentiators:

$$\begin{aligned}(\beta_{aA} + \alpha_{aA}S)\dot{a} + (\gamma_{bA} + \beta_{bA}S + \alpha_{bA}S^2)b &= I_1 - a\gamma_{aA} \\ (\beta_{aB} + \alpha_{aB}S)\dot{a} + (\alpha_{bB}S)b &= -\gamma_{aB}a\end{aligned} \qquad (2\text{-}13)$$

Now the first row is not in the form desired. Then the first thing to do, is to start with the matrix having the highest order polynomial in the main diagonal.

D. CONCLUSIONS

The method derived here seems to be a powerful tool to handle systems containing nonlinearities, because it is possible to start with a linearized form of the equations and then add the nonlinearities as inputs to the corresponding nodes. Another advantage is that the coefficients of "S" appear separately in the block diagram which permits easy manipulation of them during the simulation; because usually (with other methods) they appear in product terms and trying to change any one of them in a complicated problem can be rather cumbersome, say in an analog simulation. This as was said, is overcome with the present block

diagram method, giving the ability of changing any one of them to see the effect in the overall response without having to change anything else. Some difficulties and restrictions will be presented in Section III.

III. SIMULATION CONSIDERATIONS

The main purpose of this section is to demonstrate some of the facts which can affect simulation, because factors other than algebraic manipulations of the equations must be considered for a valid and correct simulation. It is desired to have correct simulations not only for digital computers but also for analog computers. Three basic considerations must be discussed:

A. Stability
B. Redundant integrators
C. Algebraic loops

A. STABILITY

It is assumed that the system for which the simulation is required is known in this respect. This of course can be checked with the tools given by basic control theory, e.g., Routh, Nyquist, Bode. Needless to say that an unstable system will give unstable solutions regardless of programming skill and the only way to obtain a stable solution is by altering the characteristic equation. This is a topic related with compensation and is not considered here. When working with simultaneous equations, the number of active elements can affect the stability of the simulation. Following the treatment given in Ref. 9, an example concerning an even number of active elements in a loop is given:

$$\dot{x} + ax = y \qquad (3\text{-}1)$$

$$bx = y - f \qquad \text{where f is the forcing function} \qquad (3\text{-}2)$$

Instead of substituting Eq. (3-2) in Eq. (3-1) and then simulating the resultant equation, both equations will be treated separately:

$$\dot{x} = y - ax$$

$$y = f + bx$$

An analog representation would be:

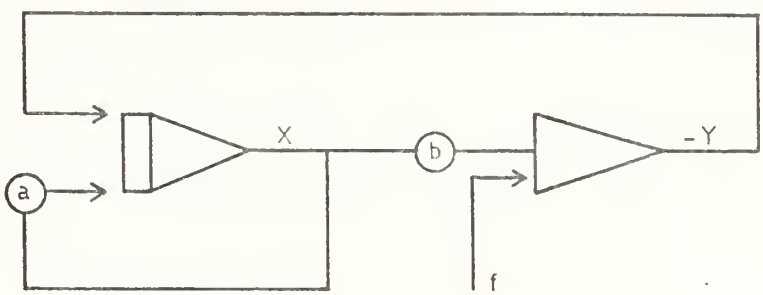


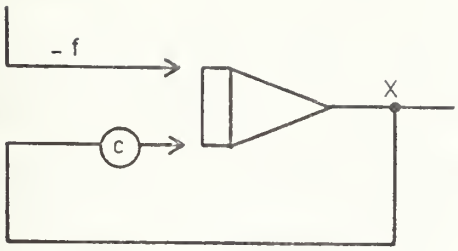
Figure 3-1a Analog diagram Eqs. (3-1) and (3-2)

There are an even number of amplifiers in the configuration of Fig. 3-1a. Substituting Eq. (3-2) into Eq. (3-1):

$$\dot{x} = f + bx - ax \qquad \text{or:}$$

$$\dot{x} = (b-a)x + f$$

The root of the characteristic equation is $S = (b-a)$. Then the computer program is stable if $b<a$.



if b<a let (b-a) = -c

$$\dot{x} = -cx + f$$

Figure 3-1b Stable case

Obviously the system is stable and only one amplifier is required.

If b>a let (b-a) = c

and $\dot{x} = cx + f$

an inverter is required:

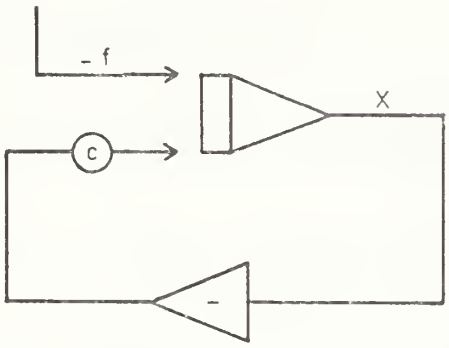


Figure 3-1c Unstable case

then two amplifiers are needed for the unstable case.

B. REDUNDANT INTEGRATORS

In dealing specially with analog simulation, a system which is stable can generate erroneous or unstable solutions

if the number of integrators used exceeds the degree of the characteristic equation. This ideally should not occur but in practice the fact that the integrators are not exact can create undesired roots which effectively change the characteristic equation. Such consideration is not quite rigidly applicable to digital simulation, but must be kept in mind.

It must be said at this stage that in the development in previous sections there appears to be an excess in the number of integrators used. This is not quite so, because the diagrams show only an algebraic result. Taking for instance Fig. 2-14 which is reproduced here in Fig. 3-2:

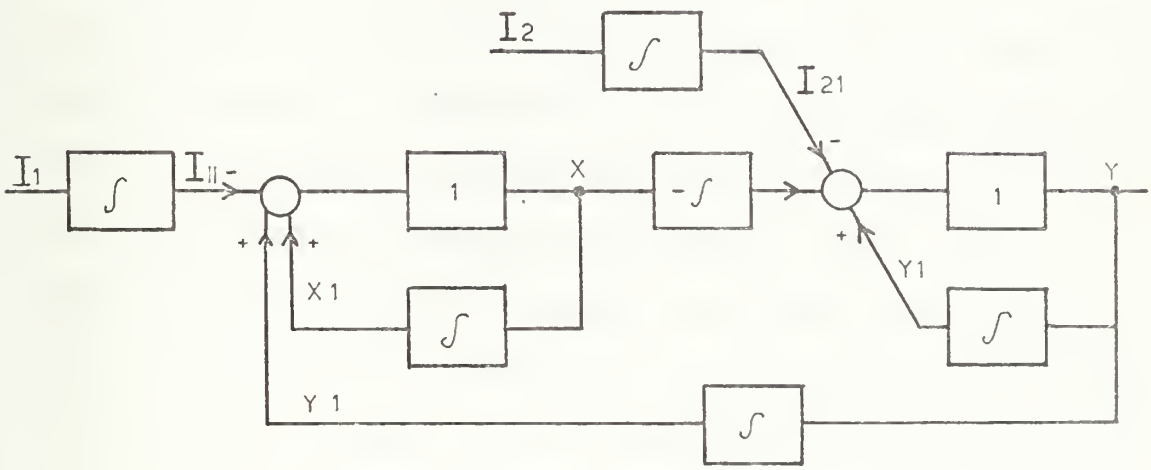


Figure 3-2 Block diagram Eq. (2-9)

A realistic interpretation of this diagram is shown in Figure 3-3a. It must be noted that the order of the characteristic equation is 2; in the diagram of Fig. 3-2 there appear to be 4 integrators in the loop, but looking at Fig. 3-3a it is shown there are only 2, as should be to correspond to the characteristic equation. The integrators in the forcing functions do not affect the solution.

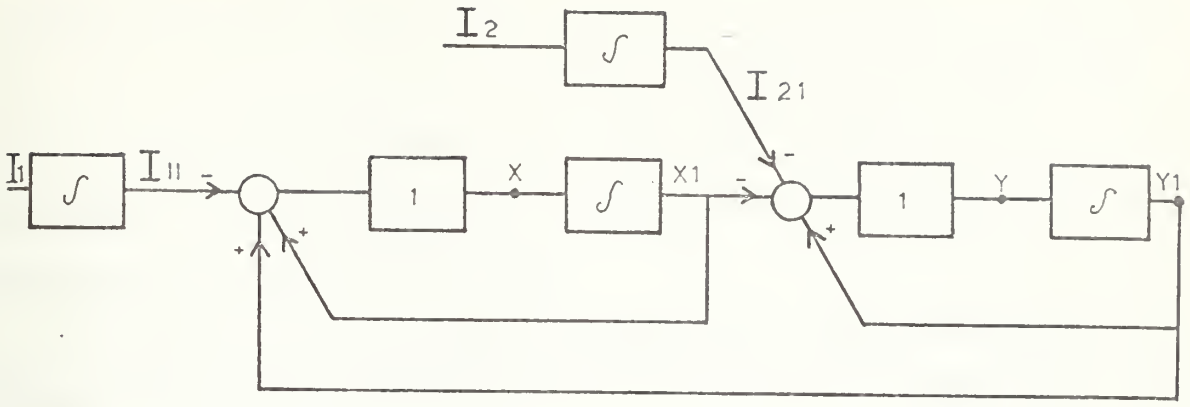


Figure 3-3a Block Transformation of Fig. 3-2

C. ALGEBRAIC LOOPS

The topic of algebraic loops or implicit loops is perhaps the more interesting and also the most difficult to visualize. By definition an implicit loop is one in which there are no integrators or other storage elements. Figure 2-12 shows the loops indicated by dotted lines. The problem mainly arises when working with a set of simultaneous equations in which the order of the highest derivative of all the variables in each particular equation is the same, see Eq. (2-7). The problem can be visualized solving for the highest order derivative of one of the variables in each equation which must be fed directly to form the highest order derivative of the other equations in the simulation.

As will be seen, this problem can seriously limit the correctness of the simulation. It is necessary to break such loops in some fashion. As indicated in Ref. 6 this could be done by manipulating the equations to eliminate the

highest order derivative from one of the equations. This is done very nicely for the case of two equations but does not seem to be very attractive when there are three or six equations as is the case of the system of equations describing the dynamics of a ship.

In order to get a better insight into the problem, a simple constant multiplier is considered (as in Ref. 9):

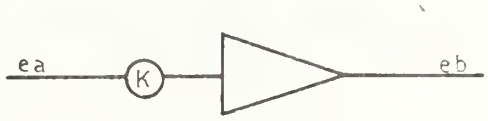


Figure 3-3b Constant Multiplier

It is seen that $eb = -Ke_a$ but since the computing components are not perfect, suppose that the actual operation is:

$$\tau\dot{e}_b + e_b = -Ke_a$$

Then, connecting two of these "imperfect" multipliers in a loop:

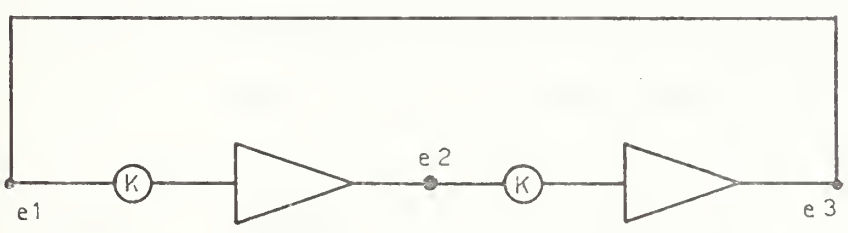


Figure 3-3c Two Multipliers in a loop

$$\tau\dot{e}_2 + e_2 = -Ke_1 \qquad (3\text{-}3)$$

$$\tau\dot{e}_3 + e_3 = -Ke_2 \quad \text{or} \quad \tau\ddot{e}_3 + \dot{e}_3 = -K\dot{e}_2 \qquad (3\text{-}4)$$

Substituting Eq. (3-4) into Eq. (3-3):

$$\frac{-\tau}{K}(\tau\ddot{e}_3 + \dot{e}_3) - \frac{1}{K}(\tau\dot{e}_3 + e_3) = -Ke_1 \qquad (3\text{-}5)$$

$$\tau^2\ddot{e}_3 + 2\tau\dot{e}_3 + e_3 = K^2e_1$$

Since $e_1=e_3$ Eq. (3-5) becomes:

$$\tau^2\ddot{e}_3 + 2\tau\dot{e}_3 + (1 - K^2)e_3 = 0$$

Using Laplace:

$$(\tau^2S^2 + 2\tau S + (1 - K^2))e_3 = 0$$

Applying Routh to this characteristic equation:

S^2	τ^2	$(1 - K^2)$
S^1	2τ	
S^0	$(1 - K^2)$	

then the loop is stable if $K^2<1$. This is a very important result by itself, because it predicts that satisfactory results will be obtained if the total gain around the loop is less than 1. When this is not satisfied the loop is unstable.

One important comment should be made here: the implicit loop and its instability are not a result of instability of the system itself. It is just a result of the way in which the equations are manipulated in order to set up the simulation.

For reference purposes a list of the maximum loop gain and the number of elements is reproduced from Ref. 9 as Table I.

TABLE I

Maximum Loop Gain and Number of Elements

Number of Nonintegrator equal multiplicative constant, active elements	Maximum "loop gain"
2	1
3	8
4	1
5	2.9
6	1

In digital simulation the presence of algebraic loops is detected when using CSMP or DSL by the sorting algorithm and will alert the user of its presence. CSMP suspends the simulation; DSL which is more sophisticated will try to solve the problem. It must be realized that in any event the solution is not an exact one. When using CSMP one way to get around the problem is to disable the sorting, declaring NOSORT. There is, however, a one-step lag due to the fact that some of the inputs are not known at the beginning of the computation cycle. There are two other options, one is declaring one of the blocks in the loop a MEMORY block with zero storage requirements, and the other

is by using an "Implicit" function. This latter is more exact because the "Implicit" function performs a number of iterations inside the loop and convergence is achieved. Computer programs using MEMORY and IMPLICIT corresponding to Fig. 3-4 are shown as Program 1 and Program 2 in the Computer printout section. Very good results were obtained for the case of two equations representing the longitudinal dynamics of the S^3 Semisubmerged ship Eq. (1-8), which simulation contained one implicit loop.

In trying to use the system of three equations which has several implicit loops, the 3 options were used in order to solve the problem. In all cases the implicit loop was eliminated but computer instability (overflow) was obtained. Checking the gains of the different implicit loops they are by far greater than 1. A program was run with the same conditions except that the gains were greatly reduced, then the overflow was eliminated.

This is a very remarkable result since it indicates that in digital simulation as well as analog not only the implicit loop must be destroyed if the sorting algorithm is to work, but that it is necessary that the gain of the loop is within the appropriate limits.

This does not mean that the theory developed in Section II is incorrect, but it is certainly limited by the restrictions discussed in the present section. Section IV shows a development which eliminates these problems.

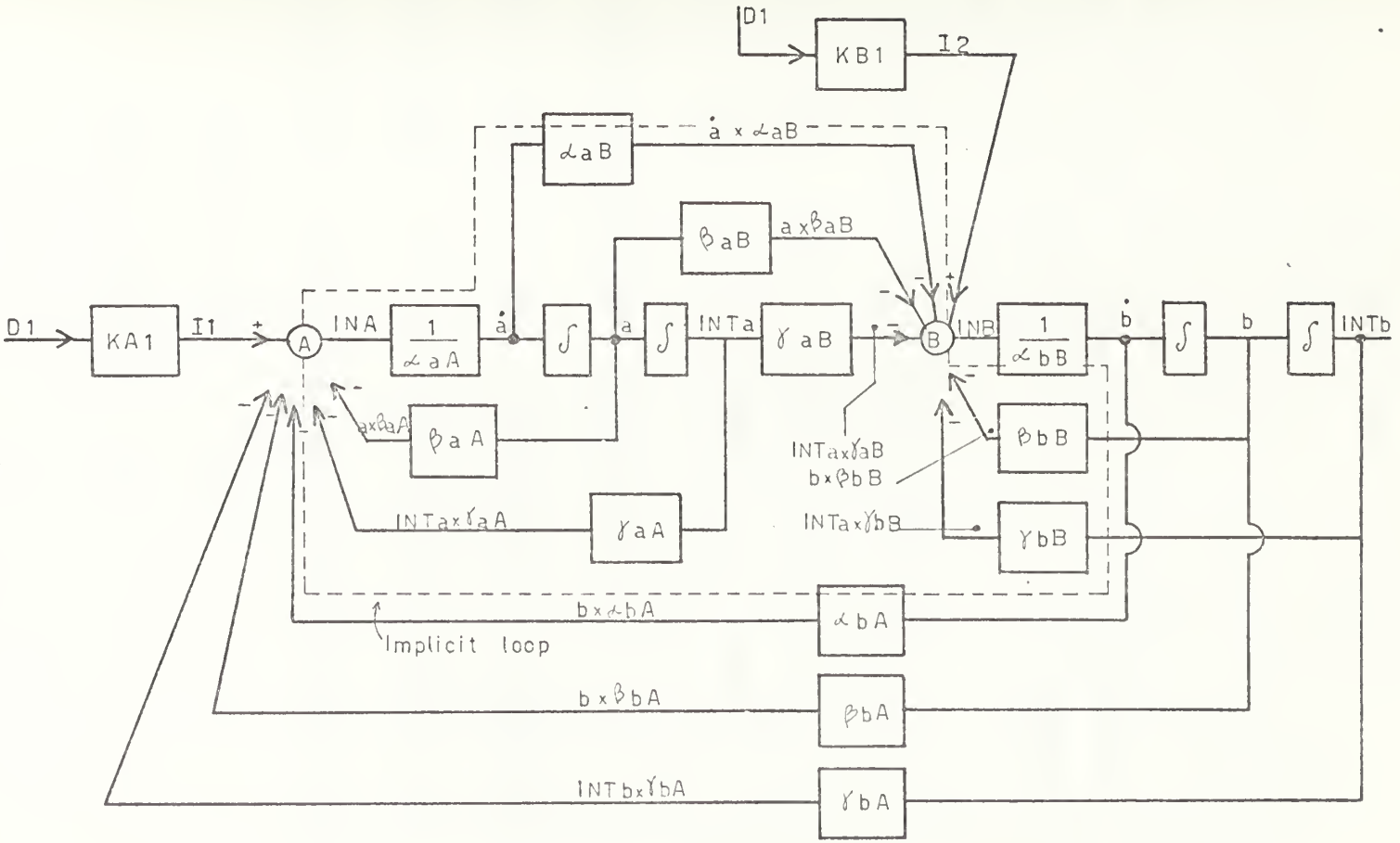


Figure 3-4 Longitudinal Dynamics of the Semisubmerged Ship (Eq. (1-8))

IV. A NEW TECHNIQUE FOR MODELING

As has been shown in Section III, if the equations are all of the same order, implicit loops will appear and no matter what is done, the simulation developed in Section II cannot be general. In dealing with the equations of motion of ships, this situation arises and a new approach will be taken.

First, suppose a system of equations:

$$(\alpha_{aA}S^2 + \beta_{aA}S + \gamma_{aA})Z + (\alpha_{bA}S^2 + \beta_{bA}S + \gamma_{bA})\theta = I_{f1}$$

$$(\alpha_{aB}S^2 + \beta_{aB}S + \gamma_{aB})Z + (\alpha_{bB}S^2 + \beta_{bB}S + \gamma_{bB})\theta = I_{f2}$$

Solving for the highest order terms

$$\begin{aligned} \alpha_{aA}S^2Z + \alpha_{bA}S^2\theta &= I_{f1} - (\beta_{aA}S+\gamma_{aA})Z - (\beta_{bA}S+\gamma_{bA})\theta \\ \alpha_{aB}S^2Z + \alpha_{bB}S^2\theta &= I_{f2} - (\beta_{aB}S+\gamma_{aB})Z - (\beta_{bB}S+\gamma_{bB})\theta \end{aligned} \qquad (4\text{-}1)$$

Calling terms in the right side I_1 and I_2 Eq. (4-1) becomes:

$$\begin{aligned} \alpha_{aA}S^2Z + \alpha_{bA}S^2\theta &= I_1 \\ \alpha_{aB}S^2Z + \alpha_{bB}S^2\theta &= I_2 \end{aligned} \qquad (4\text{-}2)$$

It is possible to solve for θ and Z:

$$\theta = \frac{\begin{vmatrix} \alpha_{aA}S^2 & I_1 \\ \alpha_{aB}S^2 & I_2 \end{vmatrix}}{\begin{vmatrix} \alpha_{aA}S^2 & \alpha_{bA}S^2 \\ \alpha_{aB}S^2 & \alpha_{bB}S^2 \end{vmatrix}} = \frac{I_2\alpha_{aA}S^2 - I_1\alpha_{aB}S^2}{\alpha_{aA}\alpha_{bB}S^4 - \alpha_{aB}\alpha_{bA}S^4} \quad (4\text{-}3)$$

$$\theta = \frac{I_2\alpha_{aA} - I_1\alpha_{aB}}{(\alpha_{aA}\alpha_{bB} - \alpha_{aB}\alpha_{bA})S^2} \quad (4\text{-}4)$$

Similarly:

$$Z = \frac{\begin{vmatrix} I_1 & \alpha_{bA}S^2 \\ I_2 & \alpha_{bB}S^2 \end{vmatrix}}{\tilde{\Delta}S^4} = \frac{I_1\alpha_{bB}S^2 - I_2\alpha_{bA}S^2}{\tilde{\Delta}S^4}$$

$$Z = \frac{I_1\alpha_{bB} - I_2\alpha_{bA}}{\tilde{\Delta}S^2} \quad (4\text{-}5)$$

$$\tilde{\Delta} = \begin{vmatrix} \alpha_{aA} & \alpha_{bA} \\ \alpha_{aB} & \alpha_{bB} \end{vmatrix} \qquad \begin{aligned} \tilde{\text{cof}}(bB) &= \alpha_{aA} \\ \tilde{\text{cof}}(aB) &= -\alpha_{bA} \\ &\text{etc.} \end{aligned} \quad (4\text{-}6)$$

Then, θ and Z can be written:

$$\theta = \frac{I_2\ \tilde{\text{cof}}(bB) + I_1\ \tilde{\text{cof}}(bA)}{\tilde{\Delta}S^2} \quad (4\text{-}7)$$

$$Z = \frac{I_1 \tilde{cof}(aA) + I_2 \tilde{cof}(aB)}{\tilde{\Delta} S^2} \tag{4-8}$$

$$\begin{aligned} I_1 &= \left[-(\beta_{aA}S + \gamma_{aA})Z - (\beta_{bA}S + \gamma_{bA})\theta \right] + I_{f1} \\ I_2 &= \left[-(\beta_{aB}S + \gamma_{aB})Z - (\beta_{bB}S + \gamma_{bB})\theta \right] + I_{f2} \end{aligned} \tag{4-9}$$

Equations (4-7) and (4-8) can be written

$$\begin{aligned} \tilde{\Delta}\ddot{Z} &= I_1 \tilde{cof}(aA) + I_2 \tilde{cof}(aB) \\ \tilde{\Delta}\ddot{\theta} &= I_1 \tilde{cof}(bA) + I_2 \tilde{cof}(bB) \end{aligned} \tag{4-10}$$

From Eq. (4-10):

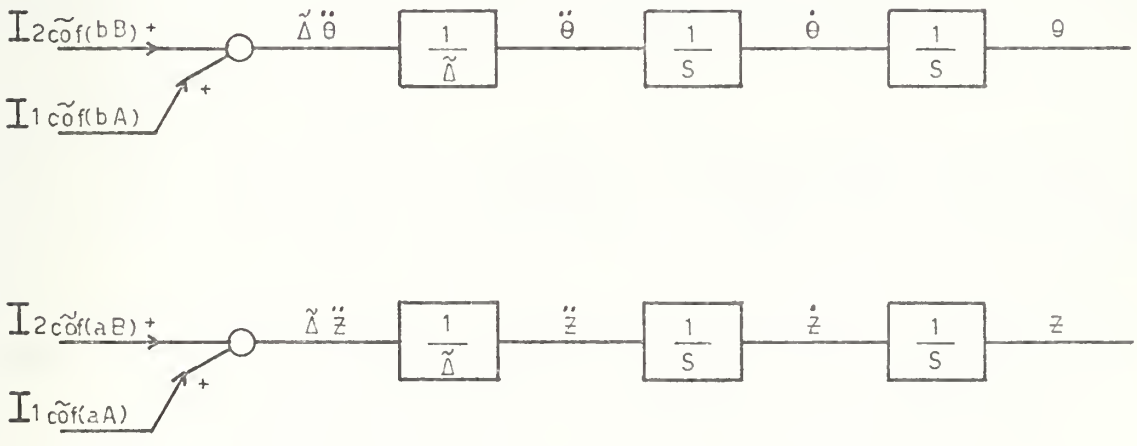


Figure 4-1 Block diagram Eq. (4-10)

All the derivatives and the functions themselves are available and it is possible to obtain I_1 and I_2. The subscript notation is preserved inside the dashed brackets of Eq. (4-9) from which, following the same criteria as before: terms above the diagonal are feedback terms, below the diagonal are forward terms. The same considerations apply to Eq. (4-10). The complete block diagram is shown in Fig. 4-2.

Once I_1 and I_2 are obtained the "linkage" of them with the rest of the circuit are just the cofactors which is done very nicely since the terms in the parenthesis indicate where to locate the block.

The block diagram does not contain any implicit loop, nor any differentiator. The present development can be extended to any number of equations but $\tilde{\Delta}$ becomes more complicated. For a system with three equations the block diagram is very easy to draw since in this case:

$$\tilde{\Delta} = \begin{vmatrix} \alpha_{aA} & \alpha_{bA} & \alpha_{cA} \\ \alpha_{aB} & \alpha_{bB} & \alpha_{cB} \\ \alpha_{aC} & \alpha_{bC} & \alpha_{cC} \end{vmatrix} = \alpha_{aA}\alpha_{bB}\alpha_{cC} + \alpha_{aB}\alpha_{bC}\alpha_{cA} + \alpha_{aC}\alpha_{cB}\alpha_{bA} - (\alpha_{cA}\alpha_{bB}\alpha_{aC} + \alpha_{cB}\alpha_{bC}\alpha_{aA} + \alpha_{cC}\alpha_{aB}\alpha_{bA}) \quad (4\text{-}11)$$

Then, the price to pay in order to avoid implicit loops is $\tilde{\Delta}$ and the cofactors, but all are functions of the coefficients of the highest order term in the polynomials, for this particular case when the order is the same for all of them.

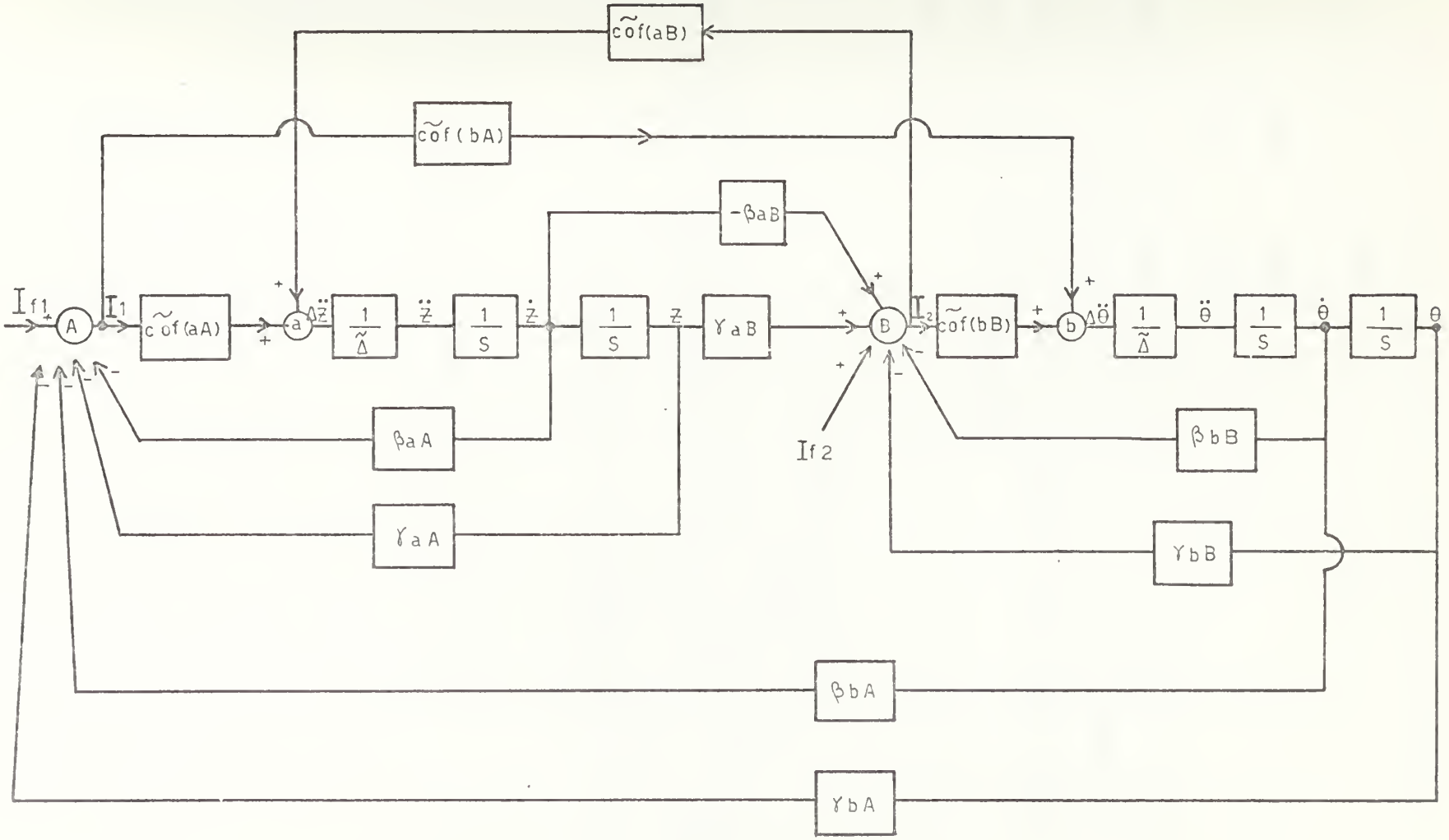


Figure 4-2 Complete block diagram Eq. (4-10)

If there is any necessity of applying this method to cases where the order is not the same, e.g.,

$$\begin{bmatrix} \alpha_{aA}S^2 & \alpha_{bA}S \\ \alpha_{aB}S^2 & \alpha_{bB}S \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \end{bmatrix} \qquad (4\text{-}12a)$$

which is a case that before could not be solved without the use of a differentiator or which could be rearranged to obtain a new variable but an implicit loop would appear. The two cases are illustrated.

Case I:

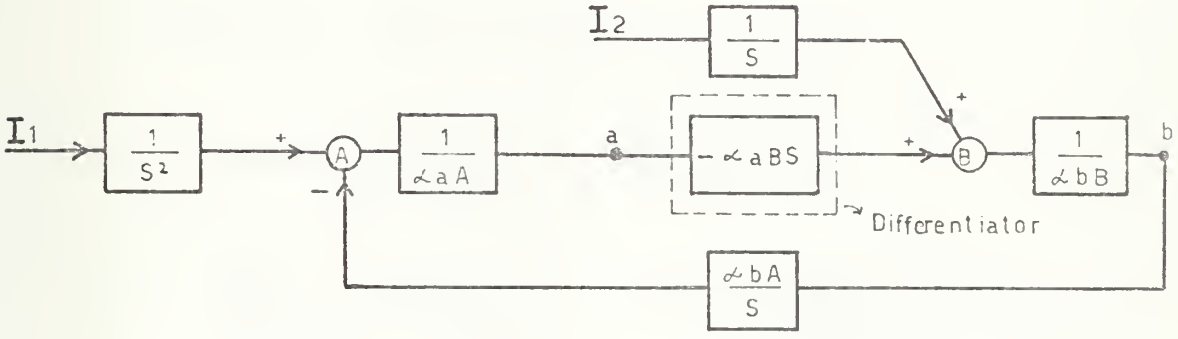


Figure 4-3 Block diagram Eq. (4-12a)

Case II:

Converting Eq. (4-12a) into:

$$\begin{bmatrix} \alpha_{aA}S & \alpha_{bA}S \\ \alpha_{aB}S & \alpha_{bB}S \end{bmatrix} \begin{bmatrix} \dot{a} \\ b \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \end{bmatrix} \qquad (4\text{-}12b)$$

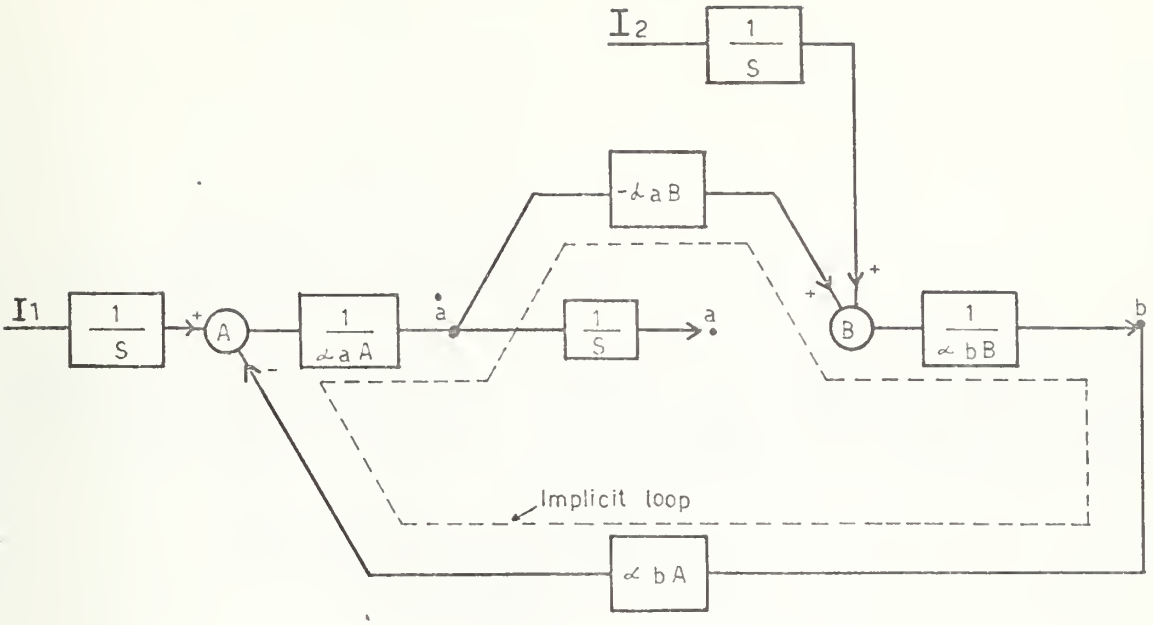


Figure 4-4 Block diagram Eq. (4-12b)

Now, applying the method just derived:

$$b = \frac{\alpha_{aA}I_2 - \alpha_{aB}I_1}{S(\alpha_{aA}\alpha_{bB} - \alpha_{bA}\alpha_{aB})} \qquad a = \frac{\alpha_{bB}I_1 - \alpha_{bA}I_2}{S^2(\alpha_{aA}\alpha_{bB} - \alpha_{bA}\alpha_{aB})}$$

$$\begin{aligned} \tilde{\Delta}\ddot{a} &= I_1\alpha_{bB} - I_2\alpha_{bA} \\ \tilde{\Delta}\dot{b} &= -I_1\alpha_{aB} + I_2\alpha_{aA} \end{aligned} \qquad (4\text{-}12c)$$

whose block diagram is shown in Figure 4-5;

where

$$I_1 = f(a,\dot{a})$$

$$I_2 = f(b)$$

or they can be functions of nonlinear terms.

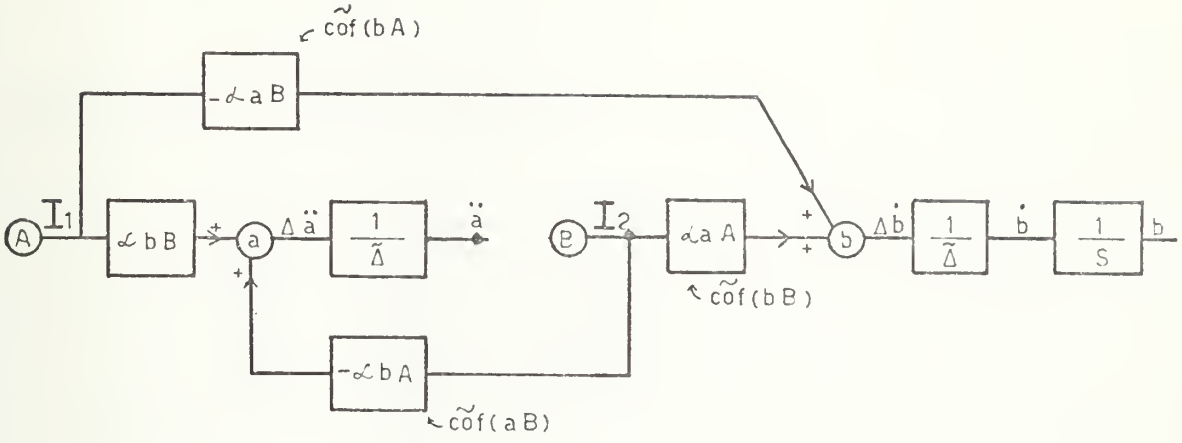


Figure 4-5 Block diagram Eq. (4-12c)

For the system of 3 equations:

$$(\alpha_{aA}S^2+\beta_{aA}S+\gamma_{aA})a + (\alpha_{bA}S^2+\beta_{bA}S+\gamma_{bA})b + (\alpha_{cA}S^2+\beta_{cA}S+\gamma_{cA})c = I_{f1}$$

$$(\alpha_{aB}S^2+\beta_{aB}S+\gamma_{aB})a + (\alpha_{bB}S^2+\beta_{bB}S+\gamma_{bB})b + (\alpha_{cB}S^2+\beta_{cB}S+\gamma_{cB})c = I_{f2}$$

$$(\alpha_{aC}S^2+\beta_{aC}S+\gamma_{aC})a + (\alpha_{bC}S^2+\beta_{bC}S+\gamma_{bC})b + (\alpha_{cC}S^2+\beta_{cC}S+\gamma_{cC})c = I_{f3}$$

Solving for highest order:

$$\begin{bmatrix}\alpha_{aA} & \alpha_{bA} & \alpha_{cA}\\ \alpha_{aB} & \alpha_{bB} & \alpha_{cB}\\ \alpha_{aC} & \alpha_{bC} & \alpha_{cC}\end{bmatrix}\begin{bmatrix}S^2a\\ S^2b\\ S^2c\end{bmatrix} = \begin{bmatrix}I_{f1}\\ I_{f2}\\ I_{f3}\end{bmatrix} - \begin{bmatrix}(\beta_{aA}S+\gamma_{aA}) & (\beta_{bA}S+\gamma_{bA}) & (\beta_{cA}S+\gamma_{cA})\\ (\beta_{aB}S+\gamma_{aB}) & (\beta_{bB}S+\gamma_{bB}) & (\beta_{cB}S+\gamma_{cB})\\ (\beta_{aC}S+\gamma_{aC}) & (\beta_{bC}S+\gamma_{bC}) & (\beta_{cC}S+\gamma_{cC})\end{bmatrix}\begin{bmatrix}a\\ b\\ c\end{bmatrix} = \begin{bmatrix}I_1\\ I_2\\ I_3\end{bmatrix} \quad (4\text{-}13)$$

$$\tilde{\Delta} = \begin{vmatrix} \alpha_{aA} & \alpha_{bA} & \alpha_{cA} \\ \alpha_{aB} & \alpha_{bB} & \alpha_{cB} \\ \alpha_{aC} & \alpha_{bC} & \alpha_{cC} \end{vmatrix} \tag{4-14}$$

Solving for "a" from Eq. (4-13):

$$a = \frac{\begin{vmatrix} I_1 & \alpha_{bA}S^2 & \alpha_{cA}S^2 \\ I_2 & \alpha_{bB}S^2 & \alpha_{cB}S^2 \\ I_3 & \alpha_{bC}S^2 & \alpha_{cC}S^2 \end{vmatrix}}{S^6\tilde{\Delta}}$$

$$a = \frac{I_1 \tilde{\text{cof}}(aA) + I_2 \tilde{\text{cof}}(aB) + I_3 \tilde{\text{cof}}(aC)}{S^2\tilde{\Delta}} \tag{4-15}$$

Note: cof( ) and $\tilde{\Delta}$ are obtained using only the coefficients of the highest order terms.

Similarly:

$$b = \frac{I_1 \tilde{\text{cof}}(bA) + I_2 \tilde{\text{cof}}(bB) + I_3 \tilde{\text{cof}}(bC)}{S^2\tilde{\Delta}} \tag{4-16}$$

$$c = \frac{I_1 \tilde{\text{cof}}(cA) + I_2 \tilde{\text{cof}}(cB) + I_3 \tilde{\text{cof}}(cC)}{S^2\tilde{\Delta}} \tag{4-17}$$

Using Eqs. (4-15), (4-16), and (4-17), the block diagram is drawn in Fig. 4-6 and the corresponding computer simulation (computer program 3), is shown in the computer printout

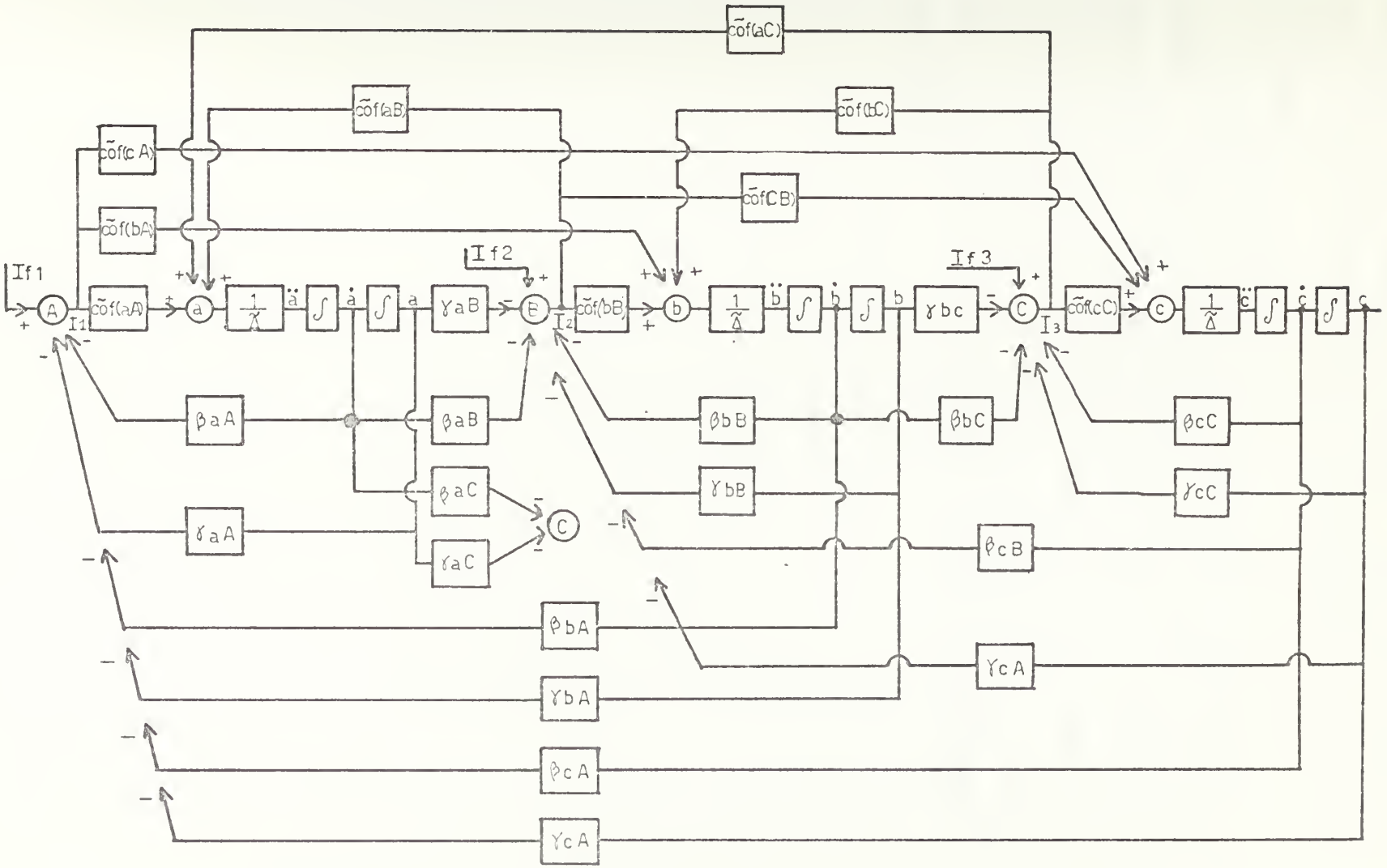


Figure 4-6 Block Diagram for System of Three Equations (Eq. 4-13)

section, for a specific example: The Lateral Dynamics of the S^3 semisubmerged ship, whose equations can be found in Section V.

As was said before, the complexity increases as the number of equations increases. For the case of a ship in six degrees of freeedom (six equations), the above results can be extended but a block diagram is very complicated and it is easier just to extend the mathematics. This will be done in the next section where the digital program will be explained.

V. COMPUTER SIMULATION

In this section the results of Section IV (equation (4-13) and Fig. 4-6) are extended to systems of six equations. The computer program 6 is developed and is shown in the computer printout section.

A. DEVELOPMENT OF THE COMPLETE PROGRAM

The value of $\tilde{\Delta}$ and of the different cofactors is obtained by using the library subroutine DTERM which was modified to meet the requirements of the program and added as a FORTRAN subprogram to the main CSMP simulation.

The name of the subprogram is: FUNCTION VALUE (Y,I,M) where Y is the array of the coefficients of the 2nd order derivative, I and M will be explained later. In order to transfer the array from the main program to the subprogram, only the first component (AAA) is used. This procedure was employed because it was noted that the locations of all the components of the array are stored column wise in the common block set up by the translator. When in the main program the statement VALUE (AAA, I, M) is used, this is interpreted by the subprogram as the array and AAA is the first component. The array Y is then copied into another array X in order to avoid the destruction of the original array in subsequent use of the subprogram. As a consequence of what has been explained, all of the coefficients of the second derivative must be listed and in the proper order, as

indicated in Section 2-A of the program. All the other coefficients can be declared in the same section and they do not require specific order. Similarly, nondefined parameters are set to zero by the sort section.

It must be noted that in determining the value of $\tilde{\Delta}$ and of the cofactors, the INITIAL-DYNAMIC OPTION of CSMP is used. This is done in order to obtain all the values before the simulation starts, because the length of the operations involved can introduce undesirable delays which are going to affect the simulation. Then, all the computations indicated between INITIAL and DYNAMIC are done before simulation time = 0. This was possible because none of the values of the coefficients are dependent of variables but of real numbers determined by the hydrodynamic coefficients. If a situation arises in which the coefficients are somehow dependent of variables, the procedure indicated here should be modified and an array must be declared in the main program, in order to solve for $\tilde{\Delta}$ and the cofactors; but it must be remembered that a delay due to computation may be introduced. The values of the coefficients in the equations are functions of the hydrodynamic coefficients or mass and inertia which are constants for all practical purposes.

A six-by-six determinant is always solved in the subprogram. As an example of the procedure used, taking the three-by-three determinant given by Eq. (4-12) and obtaining $\tilde{\text{cof}}(aA)$ which is a two-by-two determinant. First, put "1" in the position of (aA) and $\tilde{\Delta}$ and zeros in the rest of column "a" and row "A":

$$\begin{array}{c|ccc} & a & b & c \\ A & 1 & 0 & 0 \\ B & 0 & \alpha_{bB} & \alpha_{cB} \\ C & 0 & \alpha_{bC} & \alpha_{cC} \end{array} = \alpha_{bB}\alpha_{cC} - \alpha_{cB}\alpha_{bC} = \tilde{cof}(aA) \qquad (5\text{-}1)$$

Similarly:

$$\begin{vmatrix} \alpha_{aA} & 0 & \alpha_{cA} \\ \alpha_{aB} & 0 & \alpha_{cB} \\ 0 & 1 & 0 \end{vmatrix} = \alpha_{aB}\alpha_{cA} - \alpha_{cB}\alpha_{aA} = \tilde{cof}(bC) \qquad (5\text{-}2)$$

Then, this is equivalent to obtaining the signed minor of (bC):

$$\text{Minor}(bC) = \begin{vmatrix} \alpha_{aA} & \alpha_{cA} \\ \alpha_{aB} & \alpha_{cB} \end{vmatrix} = \alpha_{aA}\alpha_{cB} - \alpha_{cA}\alpha_{aB} \qquad (5\text{-}3)$$

$$\text{cof}(bC) = (-1)^5 \text{ Minor}(bC) \qquad (5\text{-}4)$$

Now taking all the cofactors and multiplying them by the corresponding I's, equations (4-15), (4-16), (4-17) are obtained after dividing by $\tilde{\Delta}$. As can be seen, the procedure is nothing else than CRAMER'S RULE but I's are dependent of the dynamics of the simulation.

As was said, in the subprogram a six-by-six determinant is solved in every case, with the appropriate branching,

$\tilde{\Delta}$ and the cofactors are obtained. In obtaining the cofactors as was explained, a "1", and zeros are placed in the corresponding row and column, this is the purpose of I and M in the argument of VALUE, e.g. $\tilde{cof}(bC)$=VALUE(AAA,3,2). In the subprogram all the terms of the matrix X corresponding to row 3 and column 2 are set = 0 and X(3,2) = 1. For $\tilde{\Delta}$, I = 0 and the determinant is taken with all its terms.

Some "write" statements are introduced in the subprogram and the determinant array and the values are printed in the output, for checking purposes.

In some cases, especially when only one mode of operation is desired, the whole program can be used by setting first the α's of the nonused variables = 1 in section 2A. The criteria here is that the value of a determinant is not changed when 0's are placed in the row and column and a "1" in the diagonal, e.g.:

$$\begin{vmatrix} \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_4 & \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 & \alpha_9 \end{vmatrix} = \alpha_1 \begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_8 & \alpha_9 \end{vmatrix} - \alpha_4 \begin{vmatrix} \alpha_2 & \alpha_3 \\ \alpha_8 & \alpha_9 \end{vmatrix} + \alpha_7 \begin{vmatrix} \alpha_2 & \alpha_3 \\ \alpha_8 & \alpha_9 \end{vmatrix} \qquad (5\text{-}5)$$

Converting Eq. (5-5) into:

$$\begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha_1 & 0 & \alpha_2 & 0 & \alpha_3 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & \alpha_4 & 0 & \alpha_5 & 0 & \alpha_6 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & \alpha_7 & 0 & \alpha_8 & 0 & \alpha_9 \end{vmatrix} = 1 \times \begin{vmatrix} \alpha_1 & 0 & \alpha_2 & 0 & \alpha_3 \\ 0 & 1 & 0 & 0 & 0 \\ \alpha_4 & 0 & \alpha_5 & 0 & \alpha_6 \\ 0 & 0 & 0 & 1 & 0 \\ \alpha_7 & 0 & \alpha_8 & 0 & \alpha_9 \end{vmatrix}$$

$$= \alpha_1 \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \alpha_5 & 0 & \alpha_6 \\ 0 & 0 & 1 & 0 \\ 0 & \alpha_8 & 0 & \alpha_9 \end{vmatrix} + \alpha_4 \begin{vmatrix} 0 & \alpha_2 & 0 & \alpha_3 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & \alpha_8 & 0 & \alpha_9 \end{vmatrix} + (-1)^{5+1}\alpha_7 \begin{vmatrix} 0 & \alpha_2 & 0 & \alpha_3 \\ 1 & 0 & 0 & 0 \\ 0 & \alpha_5 & 0 & \alpha_6 \\ 0 & 0 & 1 & 0 \end{vmatrix}$$

$$= \alpha_1 \begin{vmatrix} \alpha_5 & 0 & \alpha_6 \\ 0 & 1 & 0 \\ \alpha_8 & 0 & \alpha_9 \end{vmatrix} + (-1)^{2+1}\alpha_4 \begin{vmatrix} \alpha_2 & 0 & \alpha_3 \\ 0 & 1 & 0 \\ \alpha_8 & 0 & \alpha_9 \end{vmatrix} - \alpha_7 \begin{vmatrix} \alpha_2 & 0 & \alpha_3 \\ \alpha_5 & 0 & \alpha_6 \\ 0 & 1 & 0 \end{vmatrix}$$

$$= \alpha_1 \begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_8 & \alpha_9 \end{vmatrix} - \alpha_4 \begin{vmatrix} \alpha_2 & \alpha_3 \\ \alpha_8 & \alpha_9 \end{vmatrix} - (-1)^{3+2}\alpha_7 \begin{vmatrix} \alpha_2 & \alpha_3 \\ \alpha_5 & \alpha_6 \end{vmatrix} \qquad (5\text{-}6)$$

Since Eqs. (5-5) and (5-6) are the same, it is possible to pick the number of variables desired from the total of 6 (a, b, c, d, e, f).

The two modes, longitudinal and lateral dynamics can be studied separately by taking one dynamic, placing 1's in the nonused α's and then taking the other dynamics. It turns out that the two modes (decoupled) can be studied in the same program. This is a very strong statement and is not readily apparent since the first question arises with respect to $\tilde{\Delta}$ due to the fact that it is different for the two modes. Starting with two simple systems of equations:

$$\begin{aligned} \alpha_1\ddot{a} + \alpha_2\ddot{b} &= I_1 \\ \alpha_3\ddot{a} + \alpha_4\ddot{b} &= I_2 \end{aligned} \qquad (5\text{-}7)$$

and

$$\begin{aligned} \alpha_5 \ddot{c} + \alpha_6 \ddot{d} &= I_3 \\ \alpha_7 \ddot{c} + \alpha_8 \ddot{d} &= I_4 \end{aligned} \tag{5-8}$$

The solution of Eq. (5-7):

$$\ddot{a} = \frac{\begin{vmatrix} I_1 & \alpha_2 \\ I_2 & \alpha_4 \end{vmatrix}}{\begin{vmatrix} \alpha_1 & \alpha_2 \\ \alpha_3 & \alpha_4 \end{vmatrix}} \qquad \ddot{b} = \frac{\begin{vmatrix} \alpha_1 & I_1 \\ \alpha_3 & I_2 \end{vmatrix}}{\begin{vmatrix} \alpha_1 & \alpha_2 \\ \alpha_3 & \alpha_4 \end{vmatrix}} \tag{5-9}$$

and the solution of Eq. (5-8):

$$\ddot{c} = \frac{\begin{vmatrix} I_3 & \alpha_6 \\ I_4 & \alpha_8 \end{vmatrix}}{\begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 \end{vmatrix}} \qquad \ddot{d} = \frac{\begin{vmatrix} \alpha_5 & I_3 \\ \alpha_7 & I_4 \end{vmatrix}}{\begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 \end{vmatrix}} \tag{5-10}$$

the two $\tilde{\Lambda}$'s are different.

Now making a unique system of equations from Eqs. (5-7) and (5-8):

$$\begin{aligned} \alpha_1 \ddot{a} + \alpha_2 \ddot{b} + 0 + 0 &= I_1 \\ \alpha_3 \ddot{a} + \alpha_4 \ddot{b} + 0 + 0 &= I_2 \\ 0 + 0 + \alpha_5 \ddot{c} + \alpha_6 \ddot{d} &= I_3 \\ 0 + 0 + \alpha_7 \ddot{c} + \alpha_8 \ddot{d} &= I_4 \end{aligned} \tag{5-11}$$

the new $\tilde{\Delta}$ becomes:

$$\tilde{\Delta} = \begin{vmatrix} \alpha_1 & \alpha_2 & 0 & 0 \\ \alpha_3 & \alpha_4 & 0 & 0 \\ 0 & 0 & \alpha_5 & \alpha_6 \\ 0 & 0 & \alpha_7 & \alpha_8 \end{vmatrix} = \alpha_1 \begin{vmatrix} \alpha_4 & 0 & 0 \\ 0 & \alpha_5 & \alpha_6 \\ 0 & \alpha_7 & \alpha_8 \end{vmatrix} - \alpha_3 \begin{vmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_5 & \alpha_6 \\ 0 & \alpha_7 & \alpha_8 \end{vmatrix}$$

$$= \alpha_1\alpha_4 \begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 \end{vmatrix} - \alpha_3\alpha_2 \begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 \end{vmatrix} \qquad (5\text{-}12)$$

The numerator of $\ddot{a}$:

$$\begin{vmatrix} I_1 & \alpha_2 & 0 & 0 \\ I_2 & \alpha_4 & 0 & 0 \\ I_3 & 0 & \alpha_5 & \alpha_6 \\ I_4 & 0 & \alpha_7 & \alpha_8 \end{vmatrix} = I_1 \begin{vmatrix} \alpha_4 & 0 & 0 \\ 0 & \alpha_5 & \alpha_6 \\ 0 & \alpha_7 & \alpha_8 \end{vmatrix} - I_2 \begin{vmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_5 & \alpha_6 \\ 0 & \alpha_7 & \alpha_8 \end{vmatrix} + 0 + 0$$

or $$I_1\alpha_4 \begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 \end{vmatrix} - I_2\alpha_2 \begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 \end{vmatrix} \qquad (5\text{-}13)$$

Note that the numerator Eq. (5-13) and denominator Eq. (5-12) of $\ddot{a}$ have a common term: $\begin{vmatrix} \alpha_5 & \alpha_6 \\ \alpha_7 & \alpha_8 \end{vmatrix}$, after cancelling

$$\ddot{a} = \frac{I_1\alpha_4 - I_2\alpha_2}{\alpha_1\alpha_4 - \alpha_3\alpha_2} \qquad (5\text{-}14)$$

and Eq. (5-14) is the same expression as Eq. (5-9).

As a conclusion it can be said that even though the values of $\tilde{\Delta}$ are different for the case of each mode separately and $\tilde{\Delta}$ obtained when considering the two sets of equations, the same solutions are obtained in both cases. This is a very good result because if the complete set of equations of motion is given, but it is desired to study the two modes, lateral and longitudinal dynamics separately, just set the coupling terms = 0.

The advantage is that everything is in the same program and the solutions are obtained with only one computer run.

B. EXAMPLE WITH THE TWIN HULLED SEMISUBMERGED SHIP (S^3)

The use of the program will be explained next with an example: the S^3 semisubmerged ship [Ref. 4] .

The equations of motion in nondimensional form are rearranged to meet the forms specified in the program and in Eq. (4-13). Also these equations are linearized and the following relations used:

$$\dot{\phi} = p \ , \ \dot{\theta} = q, \ \dot{\psi} = r \ , \ \dot{z} = -u\theta + w$$

The assumptions in Section II remain valid, and consequently with the ship running at a constant (axial) velocity only five equations define the complete system for the two separate modes:

LATERAL-DIRECTIONAL DYNAMICS

Side Force:

$$((Y_{\dot{v}}-1)\dot{v} + \frac{U^2}{2}C_{Yv}v) + (Y_{\dot{r}}\dot{r} + (\frac{U^2}{2}C_{Yr}-U)r) + (Y_{\dot{p}}\dot{p} + \frac{U^2}{2}C_{Yp}p) =$$
$$-\frac{U^2}{2}C_{Y\delta r}\delta r - \frac{U^2}{2}C_{L\delta a}\delta a - Y_w$$

Rolling Moment:

$$(L_{\dot{v}}\dot{v}+\frac{U^2}{2}C_{Lv}v) + (L_{\dot{r}}\dot{r}+\frac{U^2}{2}C_{Lr}r) + ((L_{\dot{p}}-P_x^2)\dot{p}+\frac{U^2}{2}C_{Lp}p-Z_{MR}\phi) =$$

$$- \frac{U^2}{2} C_{Y\delta r}\delta r - \frac{U^2}{2} C_{L\delta a}\delta a - L_w$$

Yawing Moment:

$$(N_{\dot{v}}\dot{v}+\frac{U^2}{2}C_{nv}v) + ((N_{\dot{r}}\dot{r}-P_z^2)\dot{r}+\frac{U^2}{2}C_{nr}r) + (N_{\dot{p}}\dot{p}+\frac{U^2}{2}C_{np}p+\frac{U^2}{2}C_{n\phi}\phi) =$$

$$- \frac{U^2}{2} C_{n\delta r}\delta r - N_w$$

$$\phi = \int_0^t p\, dt \qquad P_{x,y,z} = \text{radius of gyration}$$

LONGITUDINAL DYNAMICS

(Equations (1-8) are again reproduced)

Vertical Force:

$$((Z_{\dot{w}}-1)\ddot{z} + \frac{U^2}{2}C_{zw}\dot{z}-A_{wL}z) + (Z_{\dot{q}}\ddot{\theta} + (\frac{U^2}{2}C_{zq}+Z_wU)\dot{\theta} + (\frac{U^3}{2}C_{zw}+A_{wL}\bar{X}_{wL})\theta =$$

$$- \frac{U^2}{2} (C_{z\delta c}\delta c + C_{z\delta e}\delta e) - Z_w$$

Pitching Moment:

$$(M_{\dot{w}}\ddot{z}+\frac{U^2}{2}C_{mw}\dot{z}+A_{wL}\bar{X}_{wL}z) + ((M_{\dot{q}}-P_Y^2)\ddot{\theta} + (\frac{U^2}{2}C_{mq}+M_wU)\dot{\theta} + (\frac{U^3}{2}C_{mw}-Z_{Mp})\theta =$$

$$- \frac{U^2}{2} (C_{m\delta c}\delta c + C_{m\delta e}\delta e) - M_w$$

All coefficients and variables are nondimensional.

Y_w, L_w, N_w, Z_w, M_w are the effect of waves. Now letting:

$$\dot{v} = \ddot{a} \qquad \dot{r} = \ddot{b} \qquad \dot{p} = \ddot{c} \qquad \dot{z} = \ddot{e} \qquad \dot{\theta} = \ddot{f}$$

$$v = \dot{a} \qquad r = \dot{b} \qquad p = \dot{c} \qquad \dot{z} = \dot{e} \qquad \dot{\theta} = \dot{f}$$

$$\int v = a \qquad \int r = b \qquad \phi = c \qquad z = e \qquad \theta = f$$

The next step is to identify the different coefficients

$$\alpha_{aA} = (Y_{\dot{v}} - 1)$$

$$\alpha_{aB} = L_{\dot{v}}$$

$$\alpha_{aC} = N_{\dot{v}}$$

$$\vdots$$

$$\beta_{bA} = \frac{U^2}{2} C_{Yr} - U$$

$$\vdots$$

$$KA1 = -\frac{U^2}{2} C_{Y\delta r}$$

etc.

The computer program 4a, for the system of six equations is shown in the computer printout section (actually only five equations are used in this example, as explained). The program was run for U = 1.0, canard deflection $\delta_c = 0.1$ and rudder deflection $\delta_r = 0.1$. The values for the hydrodynamic coefficients are obtained from Ref. 4 and introduced into the program as parameters.

Plots were obtained for Yaw, roll, heave and pitch and shown in Figures 5-1, 5-2, 5-3, and 5-4.[1] If the set of

---

[1] In order to obtain this type of drawing, the Library Subroutine DRAW was called by using Computer Program 7. In Section 5 of Computer Program 4a, the statement PREPARE TIME, VARIABLE was used instead of PRTPLOT VARIABLE and OUTDEL=0.2 was inserted in TIMER.

equations had nonlinear terms they could be introduced as NA, NB, NC ... in the corresponding equation.

Studies of the effect of waves and wind can also be done. In similar form to the nonlinear terms they can be introduced in the equations.

Yaw, roll, heave and pitch responses were also obtained by first deriving the transfer function for each one as explained in Section II. The values of the coefficients of the polynomials in "S" were obtained by using FORMAC [Ref. 8] and then, using DSL the transfer function was simulated. For reference purposes the DSL program used in the solution of roll is shown as computer program 4b. The transfer function is:

$$\frac{\text{Roll}}{\delta_r} = \frac{4.2098S^4 + 5.616S^3 + 1.3335S^2 + 0.12987S}{50.1011S^6 + 135.2824S^5 + 97.2277S^4 + 20.3955S^3 + 1.772S^2}$$

A comparison of the results obtained using the different methods is shown in Table II.

TABLE II COMPARISON OF RESULTS

	Yaw (1)	Roll(3)		Heave (4)		Pitch (3)	
	Max. 10^{-1} Rad.	Max. 10^{-2} Rad.	Time Seconds	Min. 10^{-1} Units	Time Seconds	Max. 10^{-2} Rad.	Time Seconds
Transfer Function Co. Pr. 4b	3.5736	8.70642	31.0	-1.370	7.6	2.947	5.4
Memory Co. Pr. 1	(2)			-1.3709	7.6	2.9533	5.4
Implicit Co. Pr. 2	(2)			-1.3672	7.6	2.945	5.4
Program Co. Pr. 4a	3.5513	8.7195	30.6	-1.3672	7.6	2.945	5.4

(1) Increasing, measured at t = 50.
(2) Cannot be used (see SectionIII).
(3) Minimum = 0 at t = 0.
(4) Maximum = 0 at t = 0.

Co. Pr. = Computer Program
Time is nondimensional

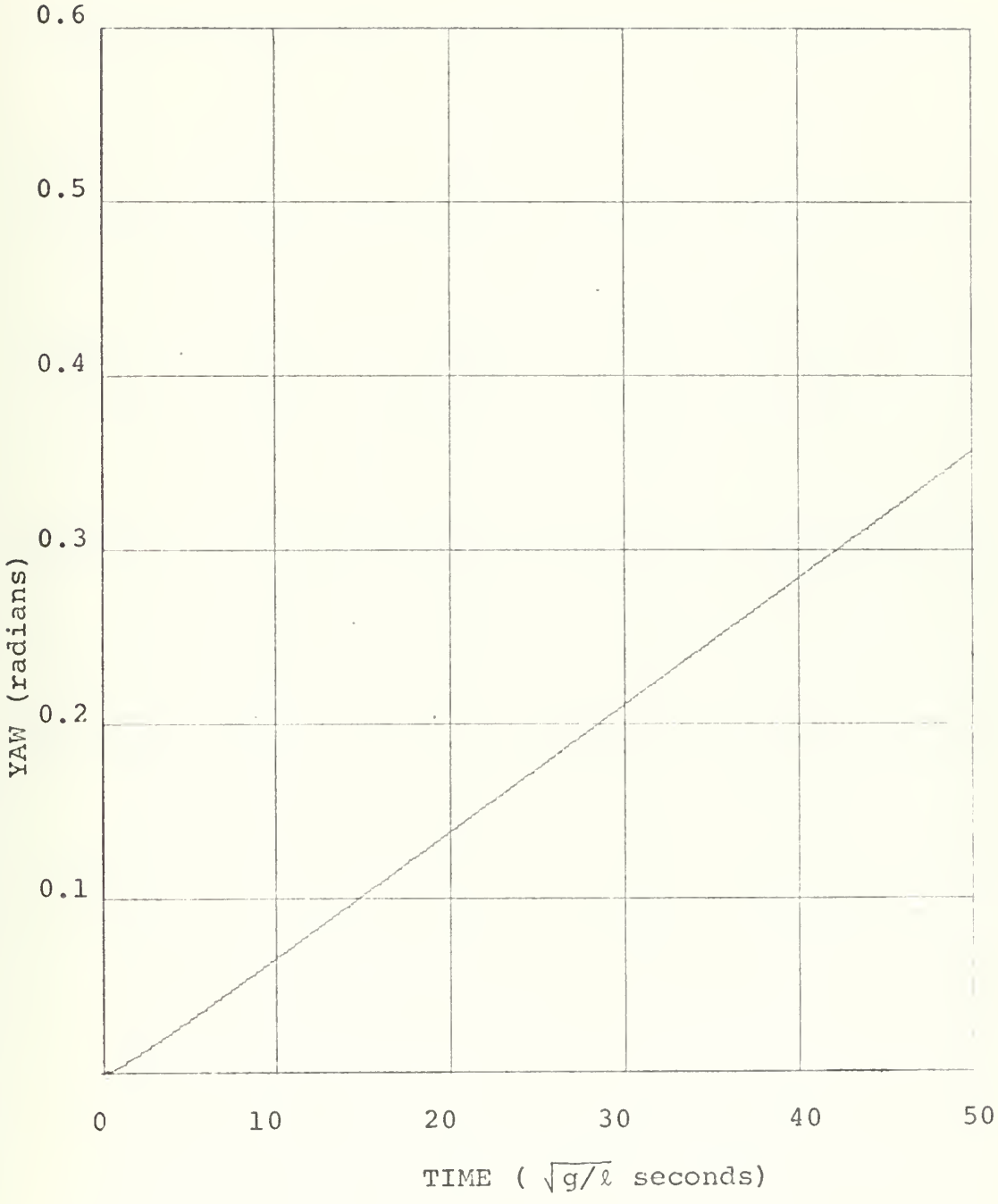


Figure 5-1 S^3 Yaw response, $U=1.0$, $\delta_r=0.1$

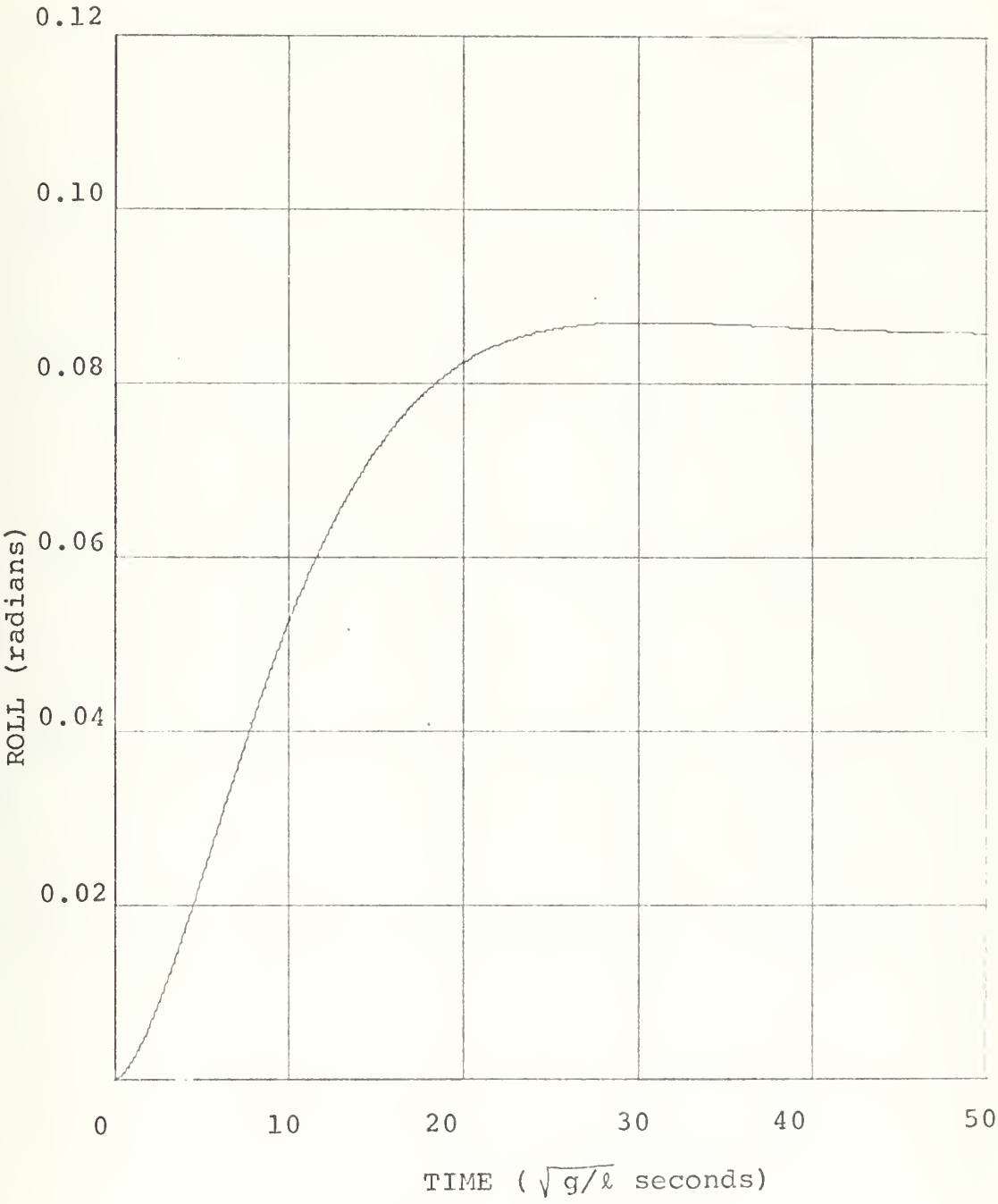


Figure 5-2 S^3 Roll response, U=1.0, δr=0.1

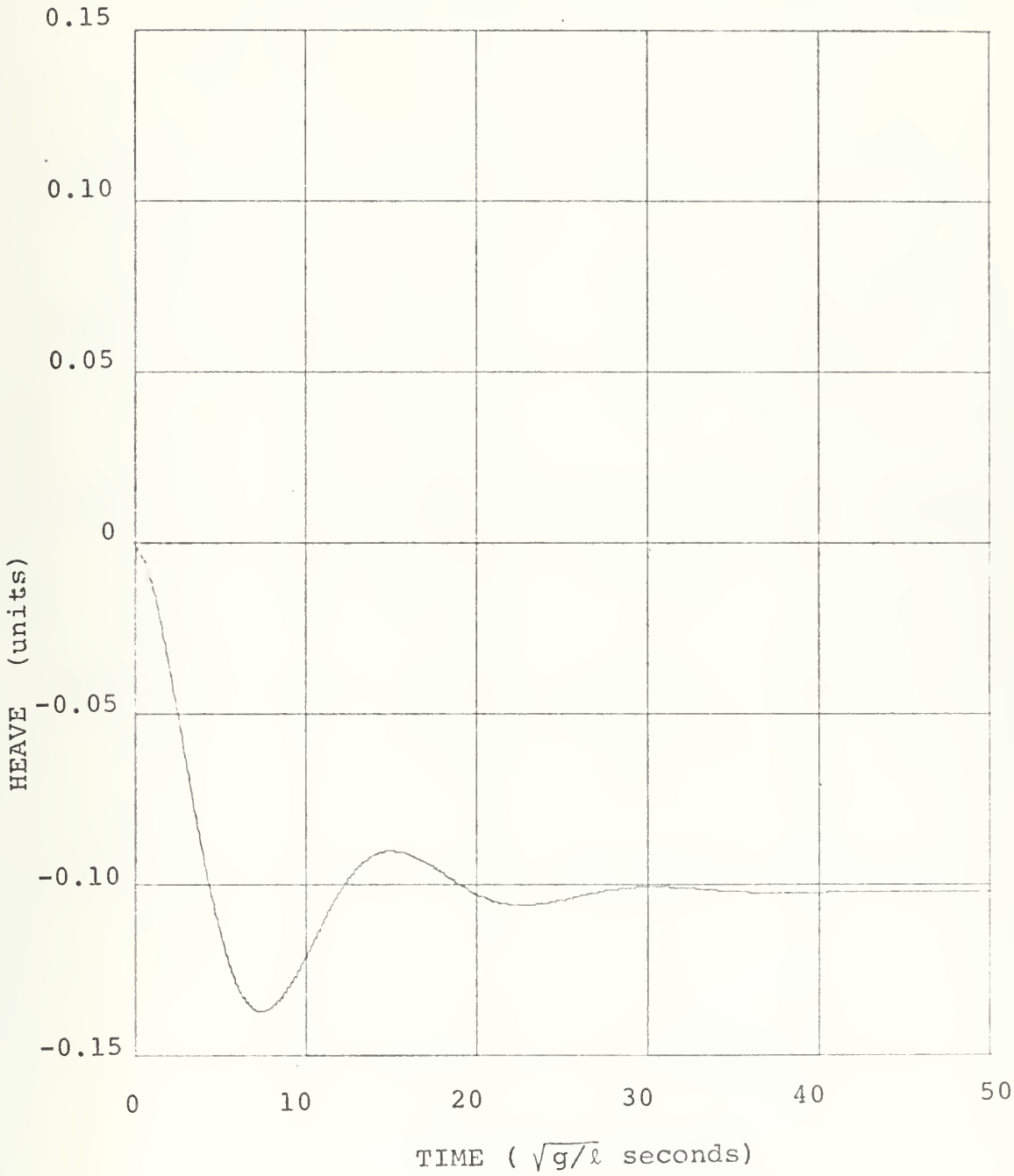


Figure 5-3 S^3 Heave response, $U=1.0$, $\delta_C=0.1$

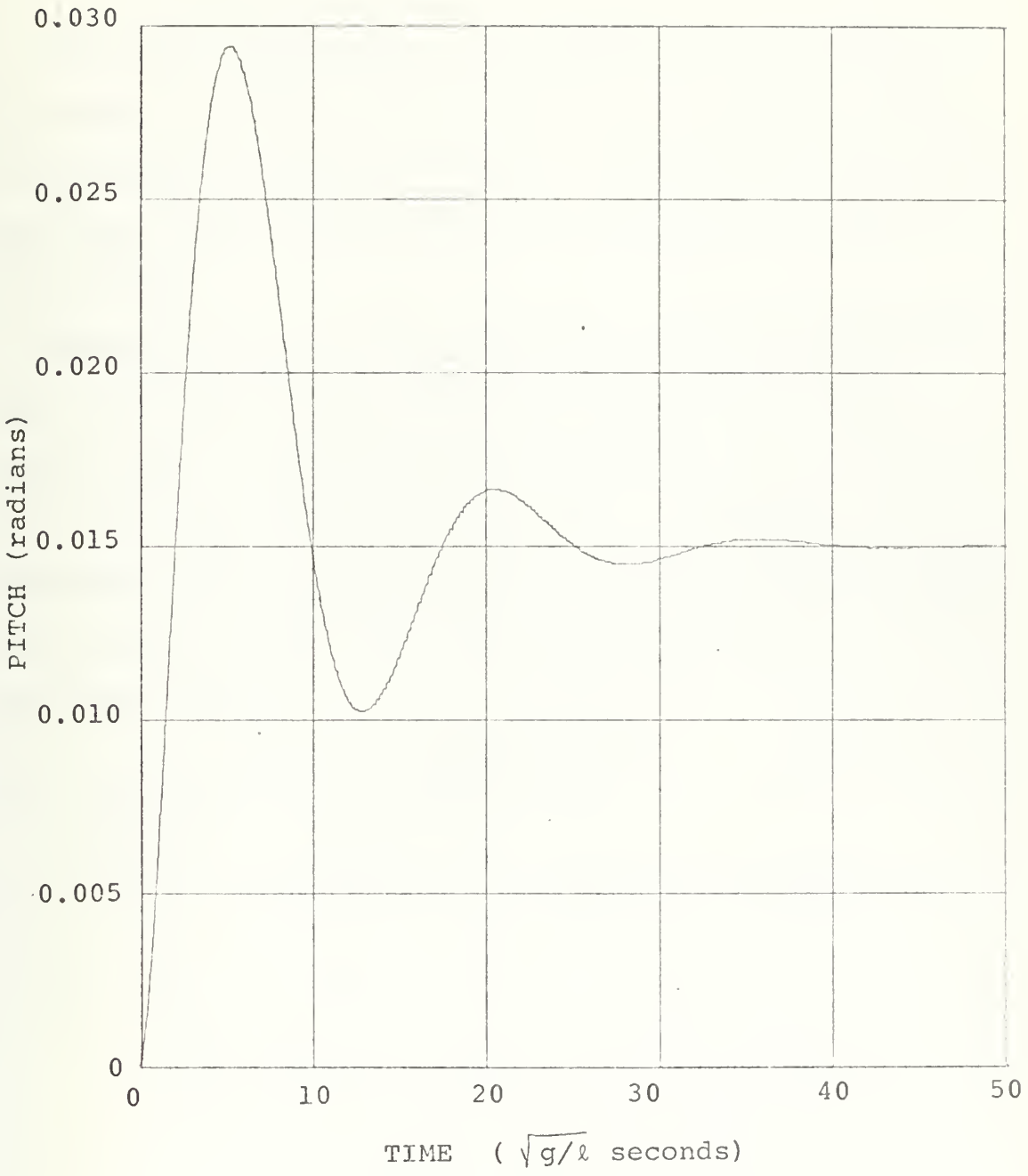


Figure 5-4 S^3 Pitch response, $U=1.0$, $\delta_c=0.1$

VI. THE SUBMARINE IN SIX DEGREES OF FREEDOM

The equations of motion of a submarine in six degrees of freedom are used in the simulation of program 5 (shown in the computer printout section). The equations are very nonlinear and all these terms are introduced as shown in the program.

The first step is the derivation of the mass and inertias as a function of ℓ, the length of the submarine. Usually this is not done since the mass and inertias are given for a specific type and length of submarine. The relations used as a function of ℓ are for a specific type of submarine but actually they are not radically different for other types.

A. DERIVATION OF MASS AND INERTIAS

$$\text{Buoyancy} = 0.518\,\ell^3$$

$$\text{mass} = \frac{\text{Buoyancy}}{\rho} = \frac{0.518}{2}\,\ell^3 = 0.259\,\ell^3$$

$$GM = 0.0635\,\ell$$

$$\text{radius of gyration (roll)} = 0.041\,\ell$$

$$\text{radius of gyration (pitch)} = 0.228\,\ell$$

$$\text{radius of gyration (yaw)} = 0.228\,\ell$$

$$I_X = \text{mass} \times (\text{radius (roll)})^2$$

$$I_X = (0.259)(0.041)^2\ell^5 = 0.0004354\,\ell^5$$

Similarly,

$$I_Y = 0.0134638\,\ell^5$$

$$I_Z = 0.0134638\,\ell^5$$

$$m' = \frac{0.259 \ell^3}{\frac{\rho}{2} \ell^3} = 0.259 \qquad (\rho = 2)$$

$$I_X' = \frac{0.0004353 \ \ell^5}{\frac{\rho}{2} \ell^5} = 0.0004354$$

$$I_Y' = 0.0134638$$

$$I_Z' = 0.0134538$$

m', I_X', I_Y', I_Z' are used in the simulation.

Now it is necessary to derive the restoring moments due to buoyancy. When the submarine is submerged the metacenter (M) and the center of buoyancy (B) become coincident and consequently GM = GB. In the standard terminology Z_B is the distance from the origin of the coordinate system to the center of buoyancy. In this discussion the origin of the coordinate system coincides with the center of gravity and the center of mass. It must be remembered that the Z axis is positive in the downward direction. Then, it is possible to write:

$$Z_B = - GM$$

since the center of buoyancy is above the center of gravity. For the submarine in cosideration:

$$Z_B = - 0.0635 \ \ell$$

and

$$\text{RIGHTING MOMENT (roll)} = B \times Z_B \times \sin\phi \times \cos\theta$$

$$\text{RIGHTING MOMENT (pitch)} = B \times Z_B \times \sin\theta$$

where ϕ is roll and θ is pitch. It must be observed that these moments are just ($B \times Z_B$) times the corresponding trigonometric function considering the three-dimensional space in which the submarine is moving. After replacing B and Z_B as functions of ℓ they are introduced in the equations. These buoyancy moments will try to restore the submarine to the correct position after any perturbation.

B. THE EQUATIONS OF MOTION OF THE SUBMARINE

Axial Force: (' indicates nondimensional)

$$\begin{aligned} m'(\dot{u}-vr+wq) &= \frac{\rho\ell}{2}(X'_{qq}q^2+X'_{rr}r^2+X'_{rp}rp) \\ &+ \frac{\rho}{2}(X'_{\dot{u}}\dot{u}+X'_{vr}vr+X'_{wq}wq) \\ &+ \frac{\rho}{2\ell}(X'_{vv}v^2) \\ &+ \frac{\rho u^2}{2\ell}(X'_{\delta r\delta r}\delta r^2+X'_{\delta s\delta s}\delta s^2+X'_{\delta b\delta b}\delta b^2) \\ &+ \frac{\rho}{2\ell}(A_1u^2+A_2uu_C+A_3u_C^2) \end{aligned}$$

Lateral Force:

$$\begin{aligned} m'(\dot{v}+ur-wp) &= \frac{\rho\ell}{2}(Y'_{\dot{p}}\dot{p}+Y'_{\dot{r}}\dot{r}+Y'_{pq}pq) \\ &+ \frac{\rho}{2}(Y'_{wp}wp+Y'_{v|r|}v|r|+uY'_{r}r+Y'_{\dot{v}}\dot{v}+uY'_{p}p) \\ &+ \frac{\rho}{2\ell}(uY'_{v}v+Y'_{wv}wv+Y'_{|v|v}|v|v) \\ &+ \frac{\rho u^2}{2\ell}(Y'_{\delta r}\delta r) \end{aligned}$$

Vertical Force:

$$m'(\dot{w}+pv-uq) = \frac{\rho\ell}{2}(Z'_{\dot{q}}\dot{q}+Z'_{rr}r^2+Z'_{rp}rp)$$

$$+ \frac{\rho}{2}(Z'_{vp}vp+Z'_{vr}vr+Z'_{w|q|}w|q|+uZ'_{q}q+Z'_{\dot{w}}\dot{w}+BDZVP\ vp)$$

$$+ \frac{\rho}{2\ell}(uZ'_{w}w+Z'_{ww}w^2+Z'_{vv}v^2+Z'_{|w|w|}w|w+Z'_{|w|u|}w|+u^2Z^*)$$

$$+ \frac{\rho u^2}{2\ell}(Z'_{\delta s}\delta s+Z'_{\delta b}\delta b)$$

Rolling Moment:

$$I'_x\dot{p}+(I'_z-I'_y)qr = \frac{\rho}{2}(K'_{\dot{p}}\dot{p}+K'_{\dot{r}}\dot{r}+K'_{qr}qr+K'_{|p|p}|p|p)$$

$$+ \frac{\rho}{2\ell}(K'_{\dot{v}}\dot{v}+uK'_{p}p+uK'_{r}r+K'_{wp}wp)-\frac{1}{\ell}(0.518x0.0635)\sin\phi\cos\theta$$

$$+\frac{\rho}{2\ell^2}(uK'_{v}v+K'_{|v|v}|v|v+K'_{vw}vw)$$

$$+ \frac{\rho}{2}\frac{u^2}{\ell^2}(K'_{\delta r}\delta r)$$

Pitching Moment:

$$I'_y\dot{q}+(I'_x-I'_z)rp = \frac{\rho}{2}(M'_{\dot{q}}\dot{q}+M'_{rp}rp+M'_{rr}r^2+BDMRP\ rp)$$

$$+ \frac{\rho}{2\ell}(M'_{\dot{w}}\dot{w}+uM'_{q}q+M'_{vr}vr+M'_{vp}vp+M'_{|w|q}|w|q)$$

$$- \frac{1}{\ell}(0.518)(0.0635)\sin\theta$$

$$+ \frac{\rho}{2\ell^2}(uM'_{w}w+M'_{vv}v^2+M'_{ww}w^2+M'_{|w|w}|w|w+uM'_{|w|}|w|+u^2M^*)$$

$$+ \frac{\rho u^2}{2\ell^2}(M'_{\delta s}\delta s+M'_{\delta b}\delta b)$$

Yawing Moment:

$$I'_z\dot{r}+(I'_y-I'_x)pq = \frac{\rho}{2}(N'_{\dot{r}}\dot{r}+N'_{\dot{p}}\dot{p}+N'_{pq}pq)$$

$$+ \frac{\rho}{2\ell}(N'_{\dot{v}}\dot{v}+uN'_p p+uN'_r r+N'_{wp}wp+N'_{|v|r}|v|r)$$

$$+ \frac{\rho}{2\ell^2}(uN'_v v+N'_{wv}wv+N'_{|v|v}|v|v)$$

$$+ \frac{\rho u^2}{2\ell^2}(N'_{\delta r}\delta r)$$

Kinematic Relations:

$$\dot{\phi} = p + \dot{\psi}\sin\theta$$

$$\dot{\theta} = q\cos\phi - r\sin\phi$$

$$\dot{\psi} = \frac{r + \dot{\theta}\sin\phi}{\cos\theta\cos\phi}$$

These equations are practically the same equations as given in Refs. 3 and 7. The water density (ρ) is assumed to be equal to 2. When the roll angle is very large the equations must be modified and terms having absolute values must be included in the equations as indicated in Ref. 3. The complete notation for all the terms in the equations is given in Ref. 3, but the basic notation can be found in the Table of Symbols. Figure 6-1 shows the positive directions of the axes, angles, linear velocity components, angular velocity components, forces and moments as defined in Ref. 3.

The computer simulation was run for an initial velocity in the X direction $u = 6$ knots, and a command speed $u_c = 6$ knots. The length of the submarine was chosen to be $\ell = 20$ feet but it could be any other length. All the deflections are step functions which naturally is not a very realistic situation but ramp or any other function can be equally used. The results are shown in Figures 6.2 to 6.12.[2]

[2]See Footnote 1 on page 74.

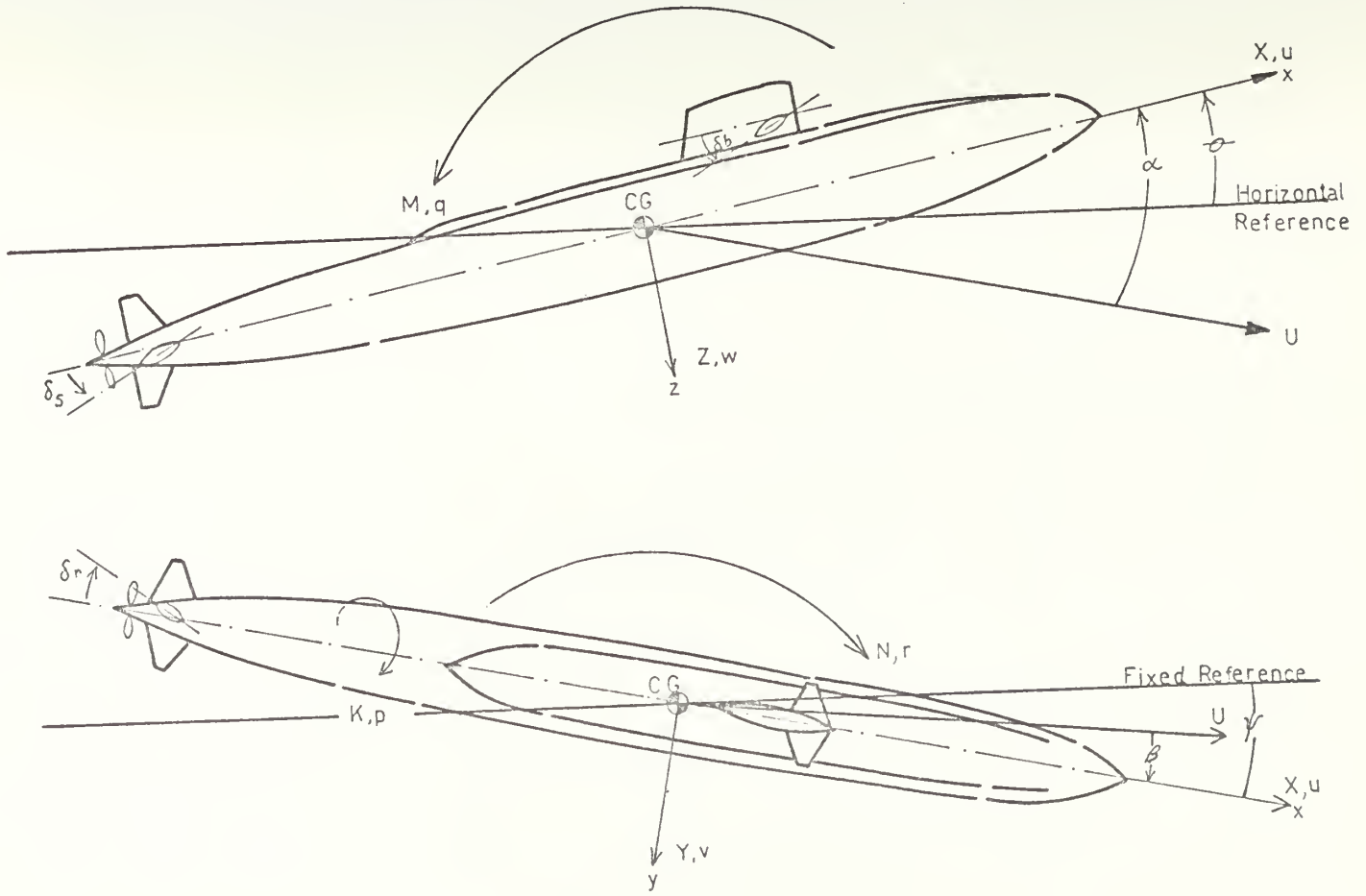


Figure 6-1 Positive Directions of Axes, Angles, Velocity Components, Forces and Moments

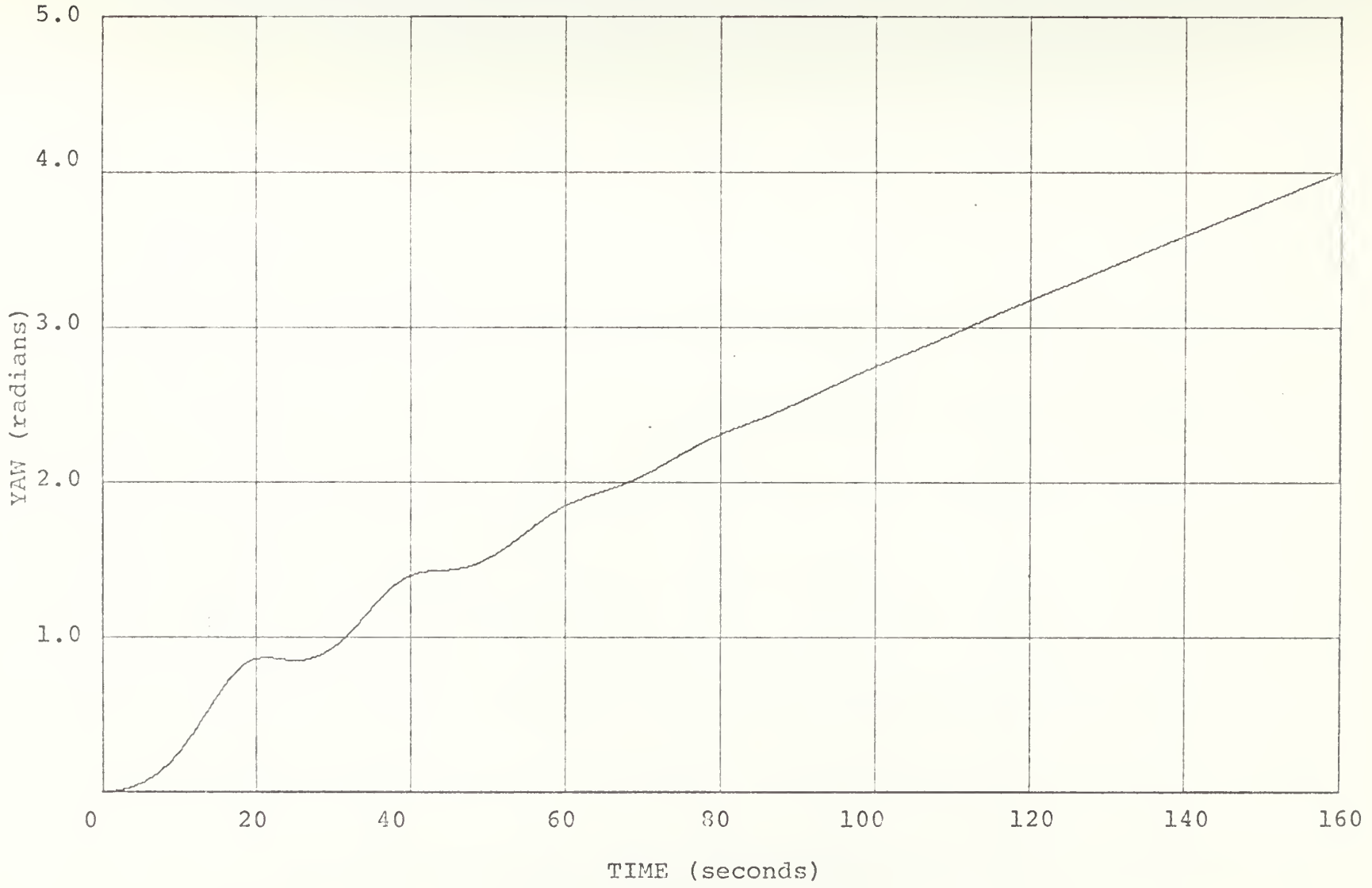


Figure 6-2 Submarine Yaw Response, δ_r=0.1, u(0)=6, u_c=6

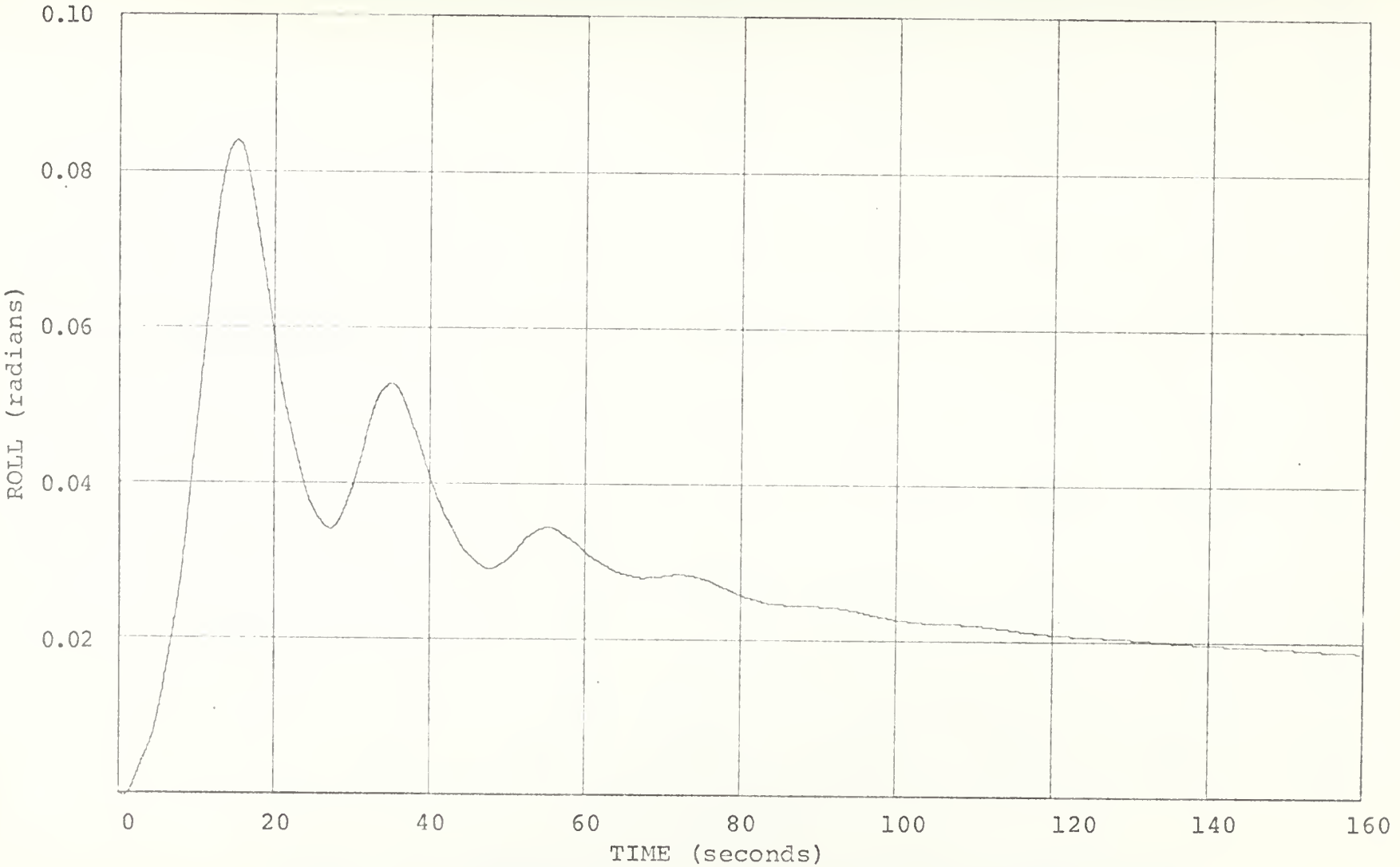


Figure 6-3 Submarine Roll Response, δ_r=-0.1, u(0)=6, u_c=6

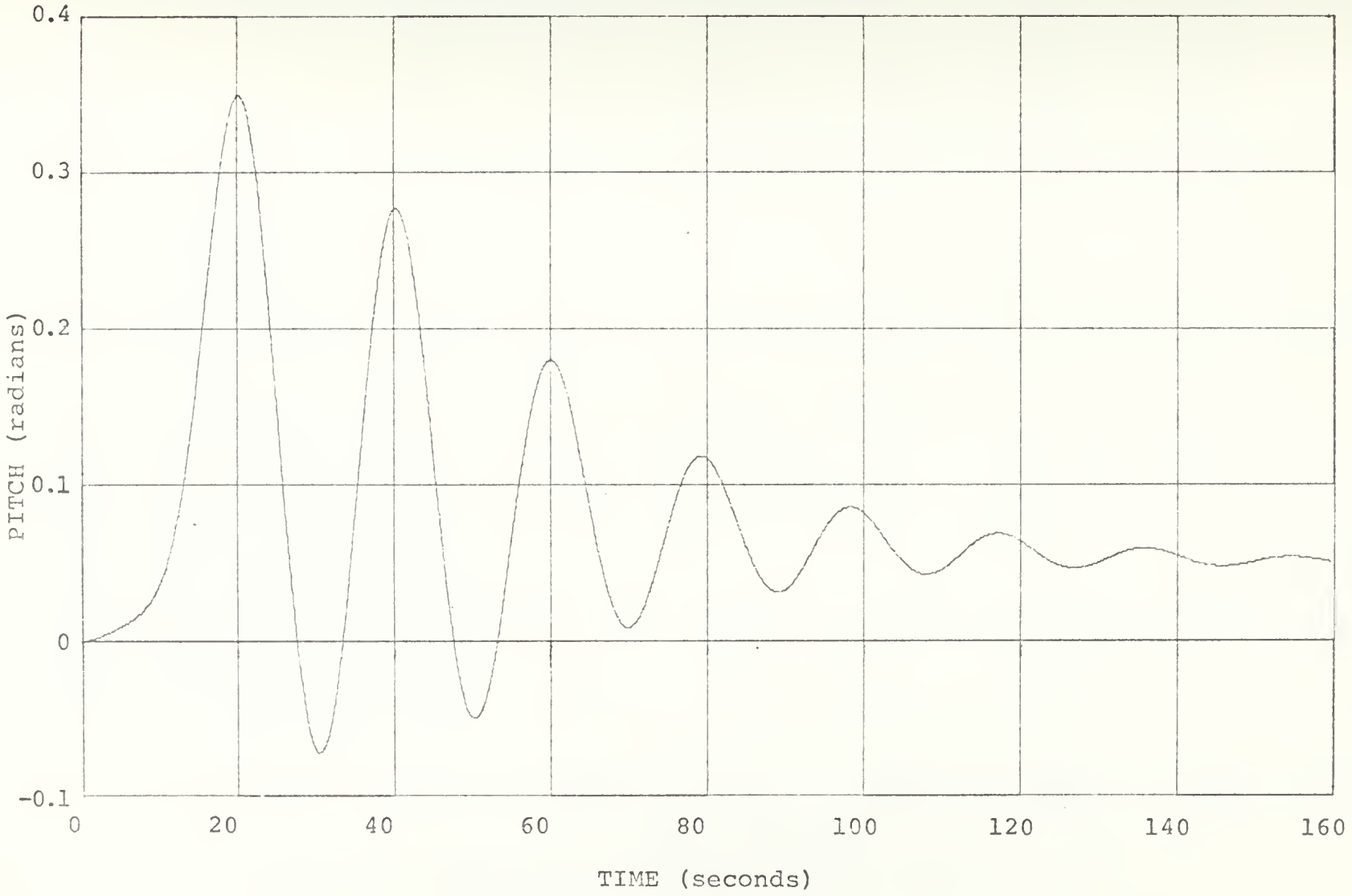


Figure 6-4 Submarine Pitch Response, δ_r=-0.1, u(0)=6, u_c=6

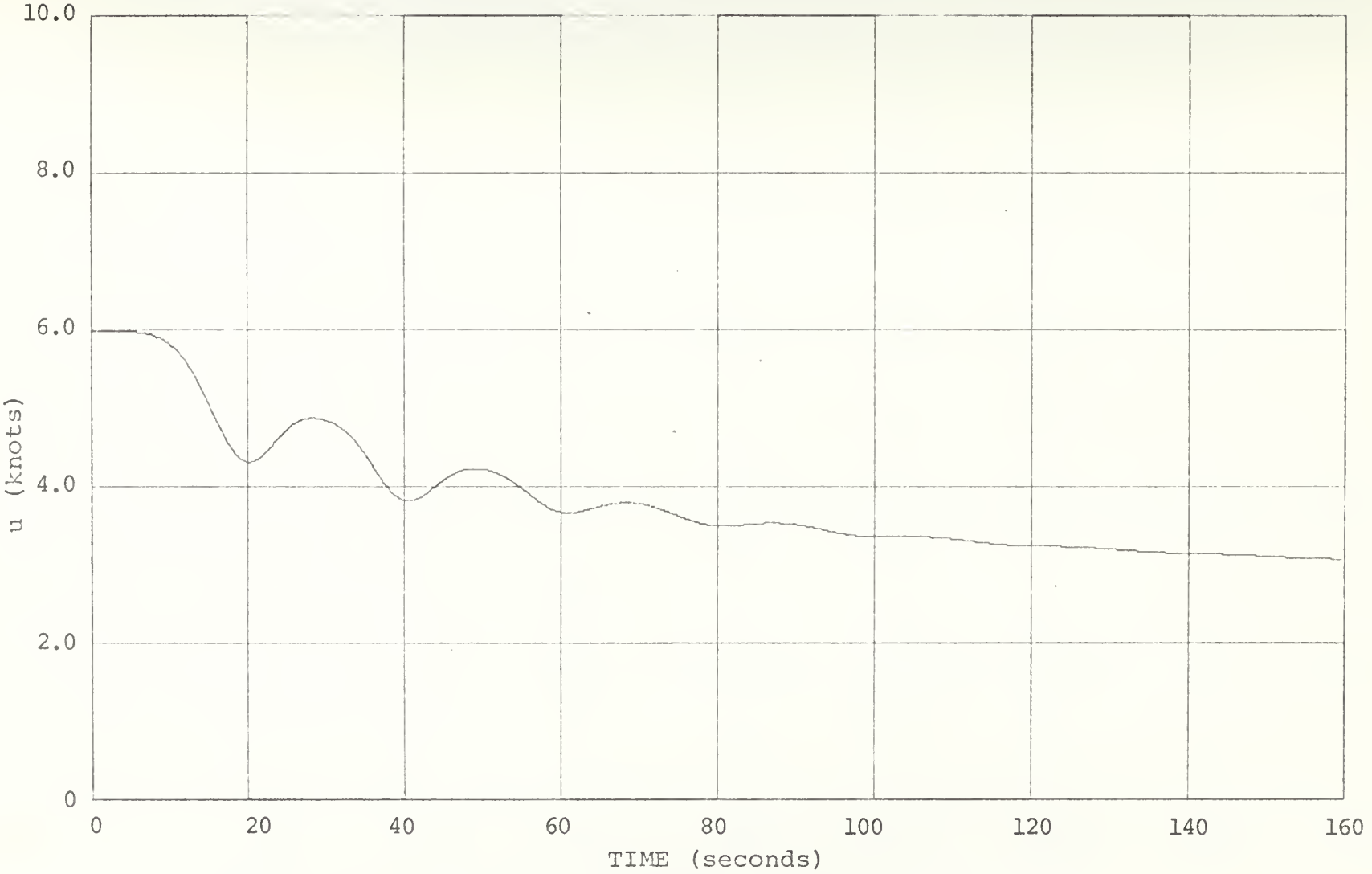


Figure 6-5 Submarine u Response, δ_r=-0.1, u(0)=6, u_c=6

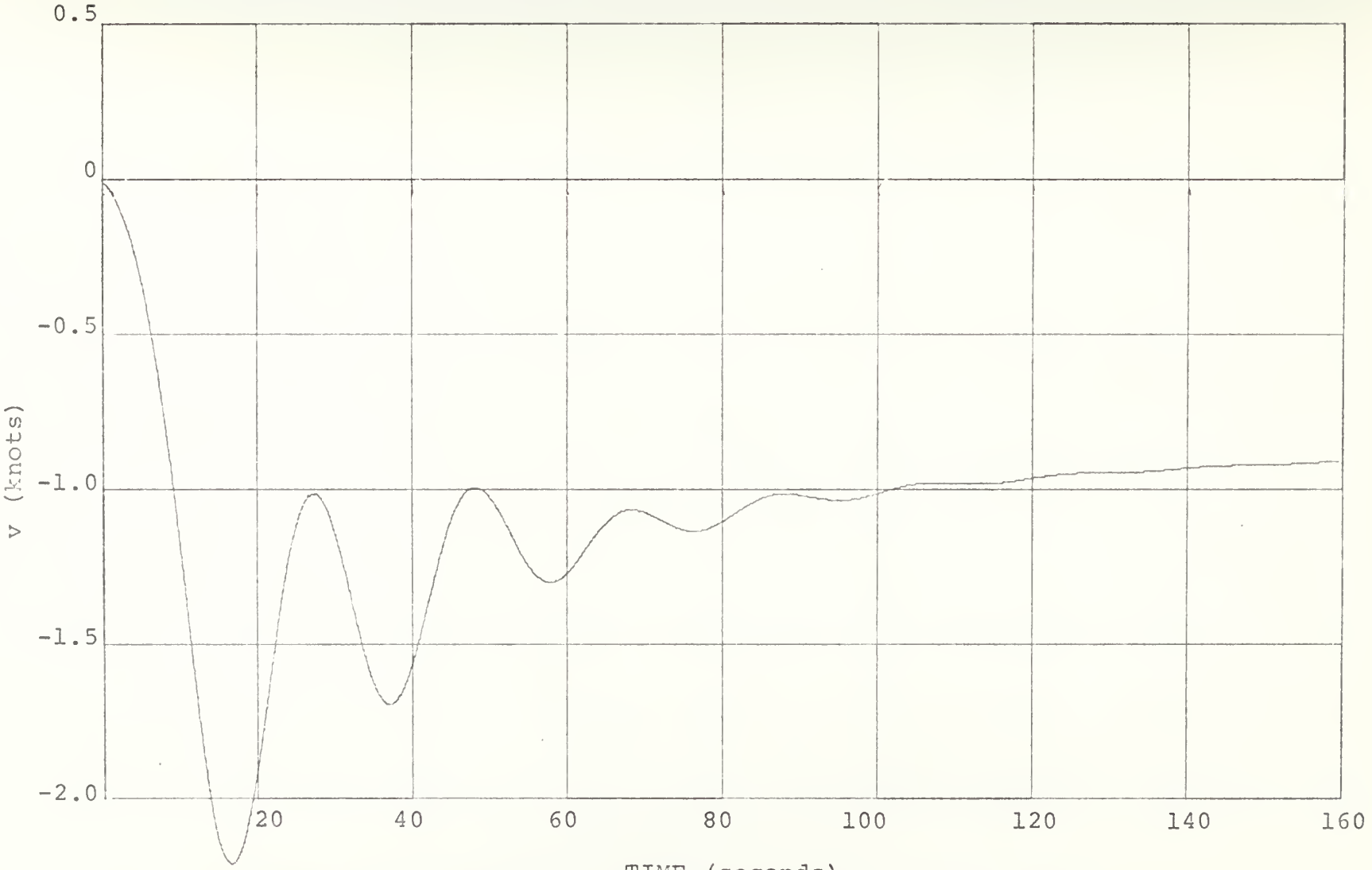


Figure 6-6 Submarine v Response δ_r=-0.1, u(0)=6, u_c=6

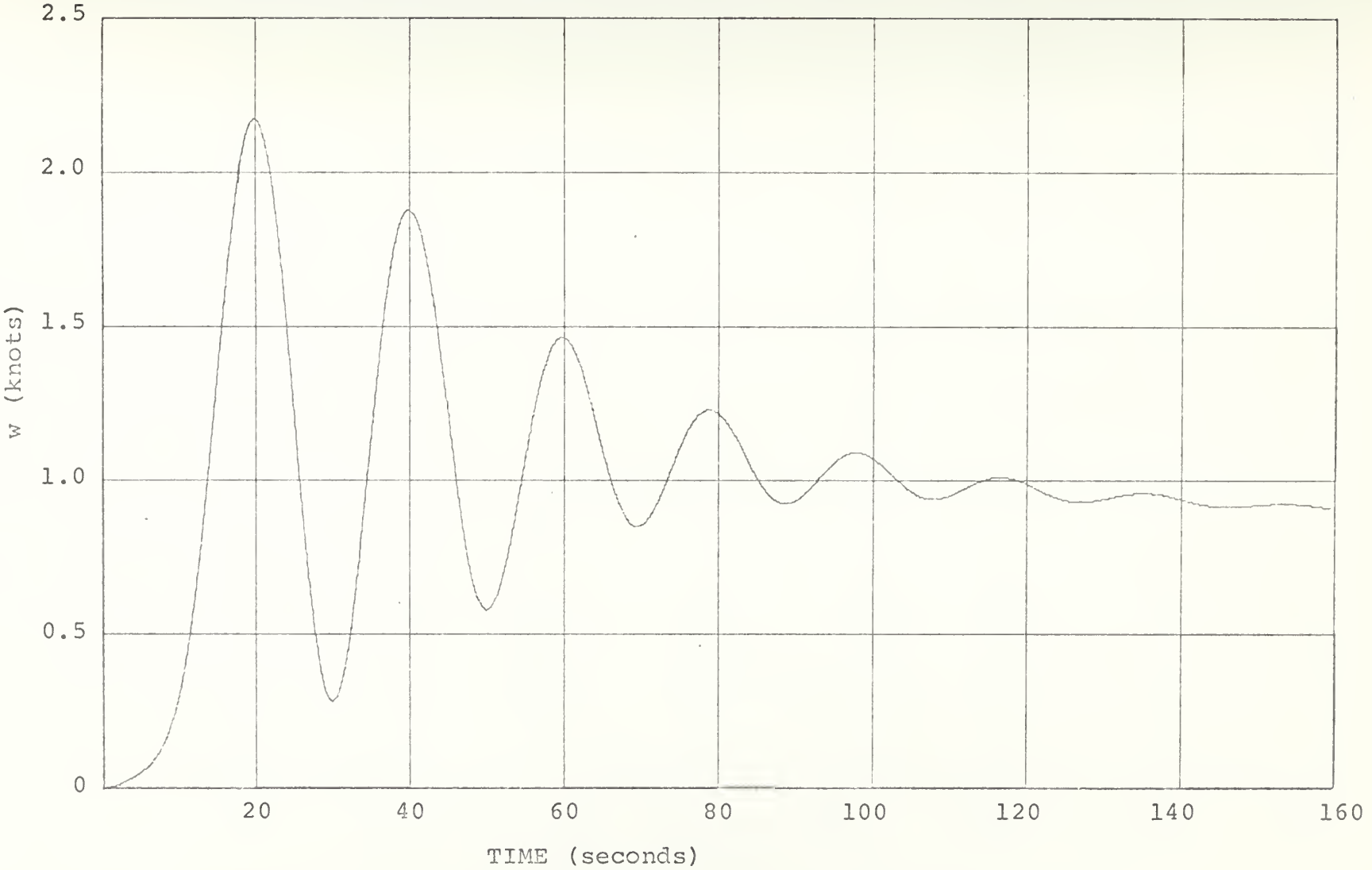


Figure 6-7 Submarine w Response, δ_r=-0.1, u(0)=6, u_c=6

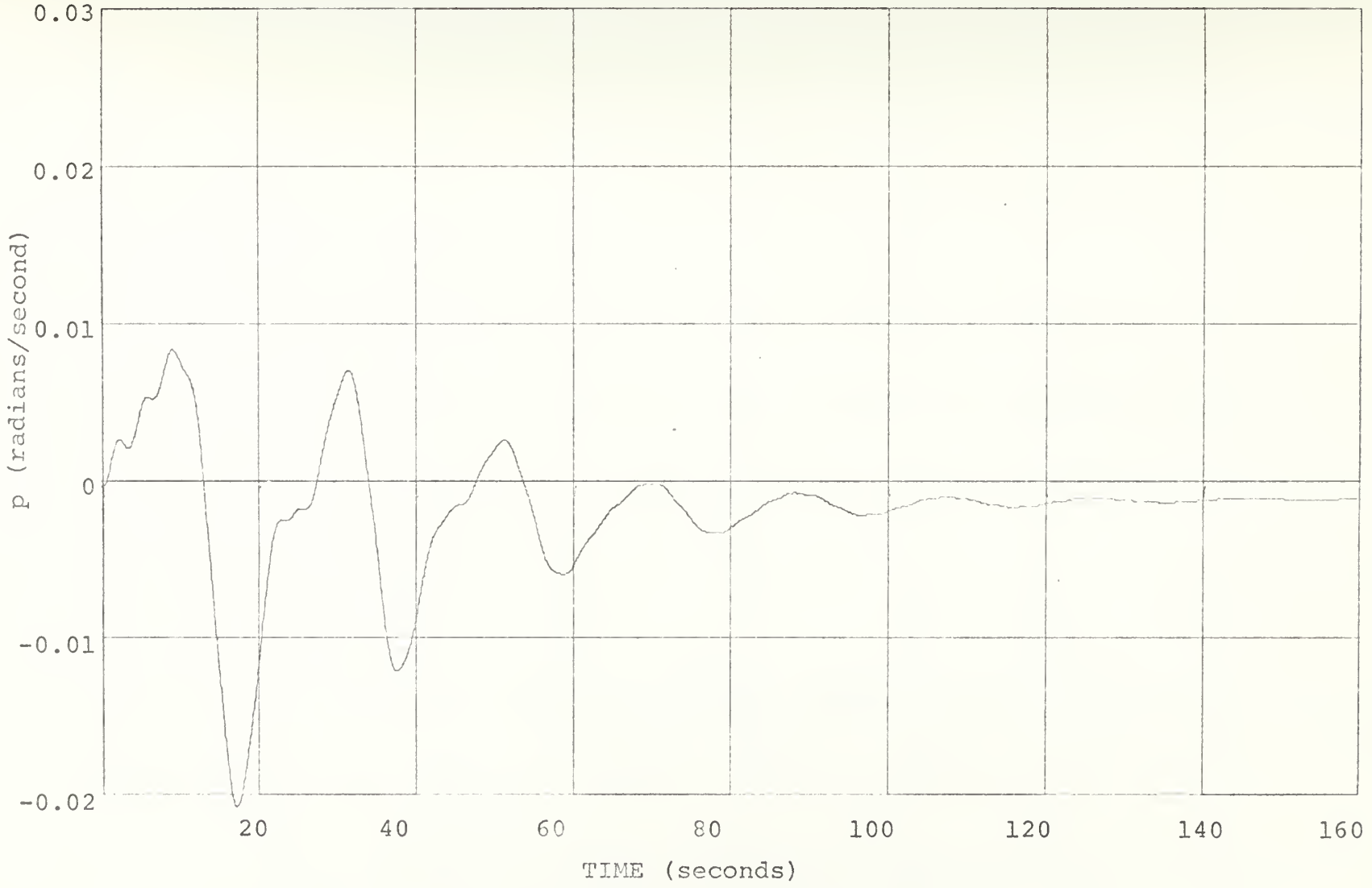


Figure 6-8 Submarine p response, δ_r=-0.1, u(0)=6, u_c=6

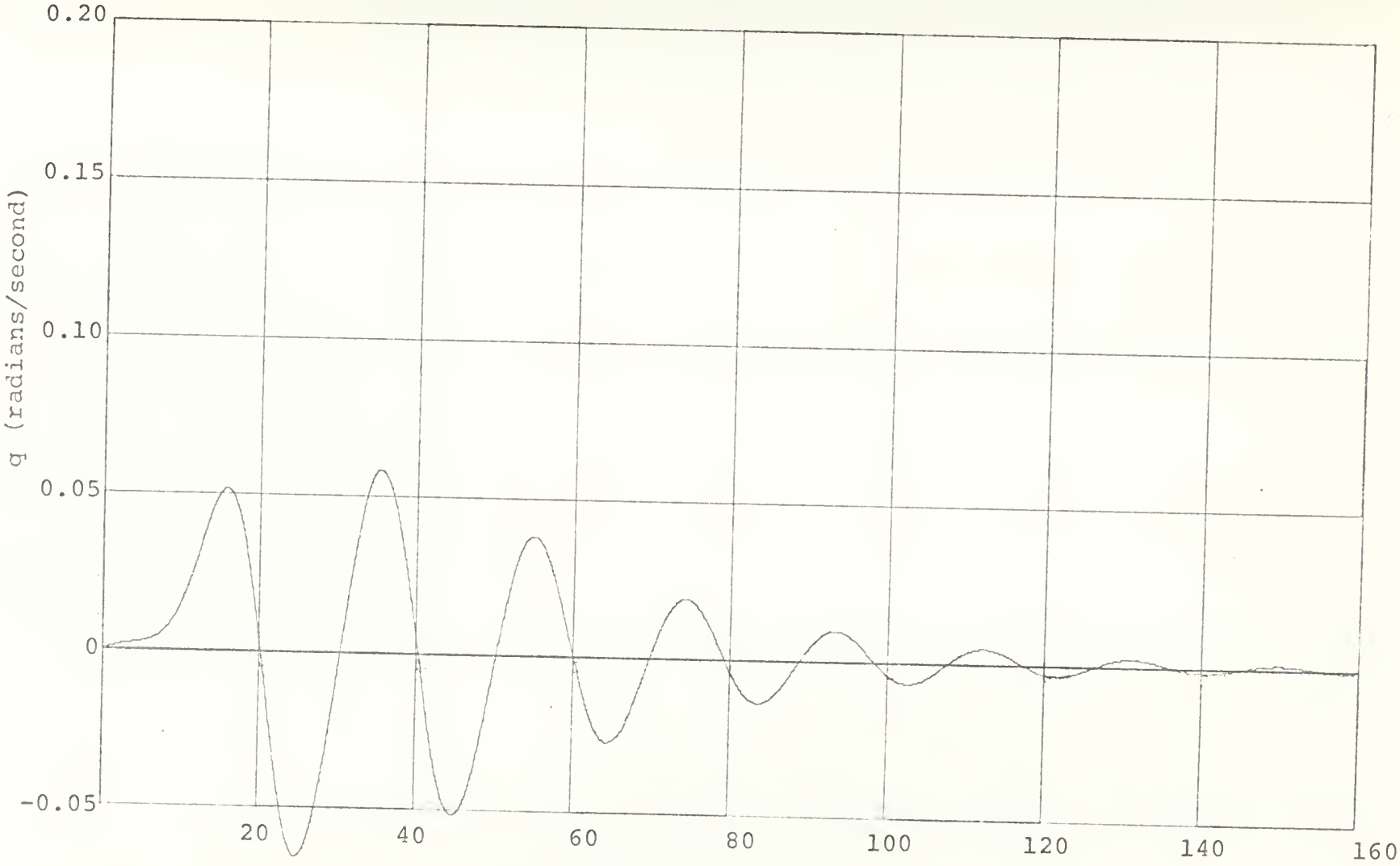


Figure 6-9 Submarine q response, $\delta r=-0.1$, $u(0)=6$, $u_c=6$

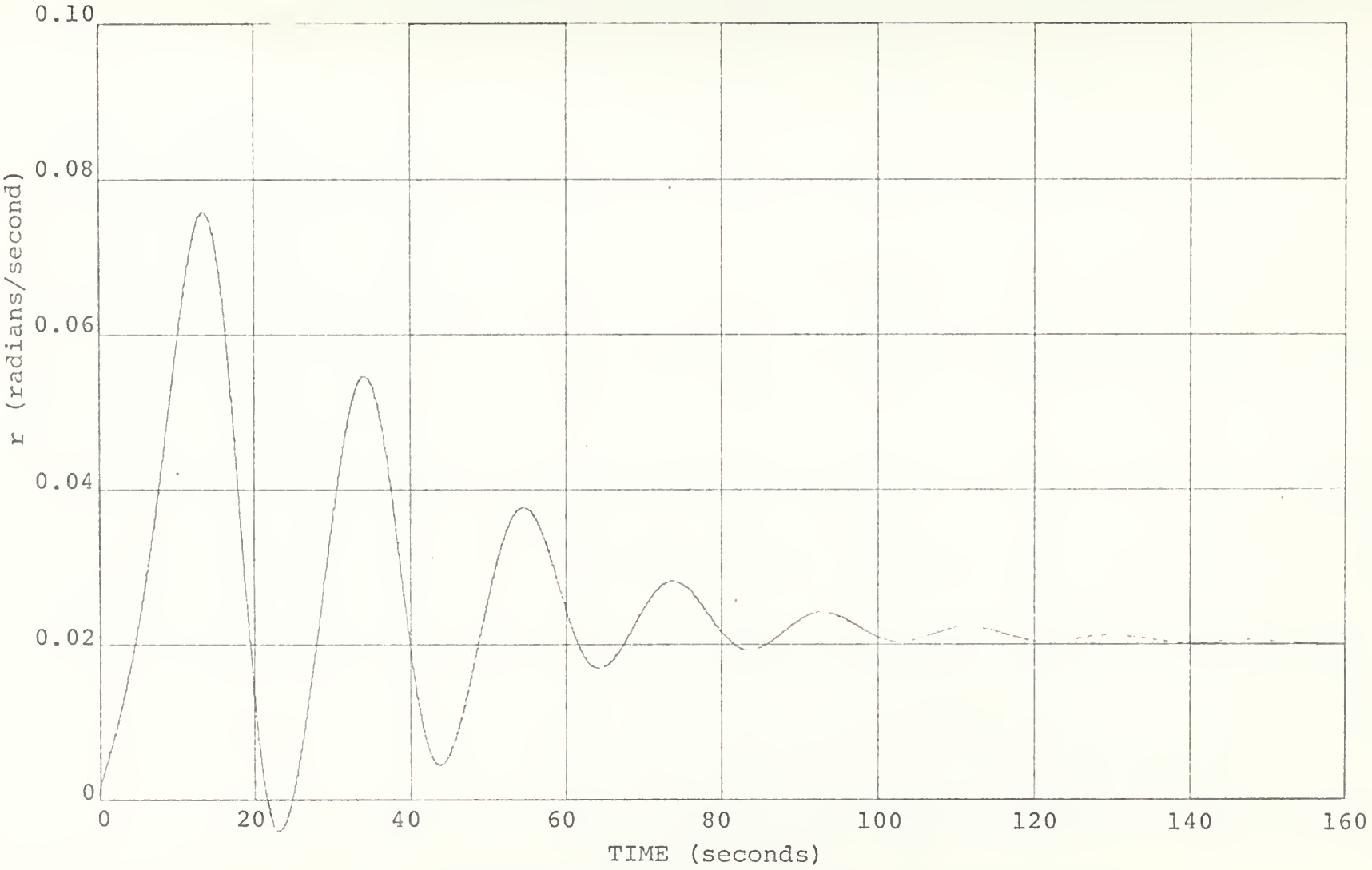


Figure 6-10 Submarine r Response, δ_r=-0.1, u(0)=6, u_c=6

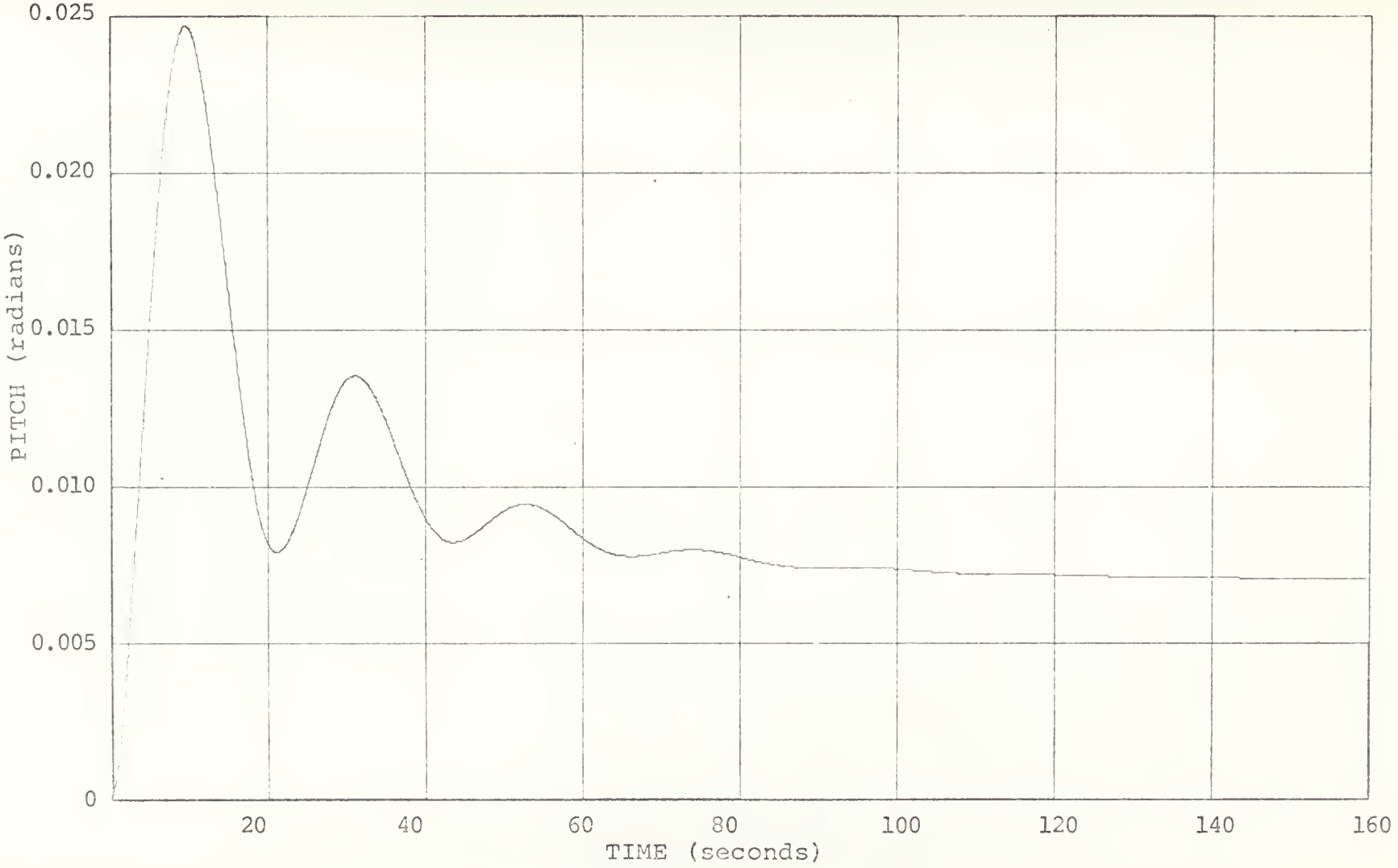


Figure 6-11 Submarine Pitch Response, δ_b=0.1, u(0)=6, u_c=6

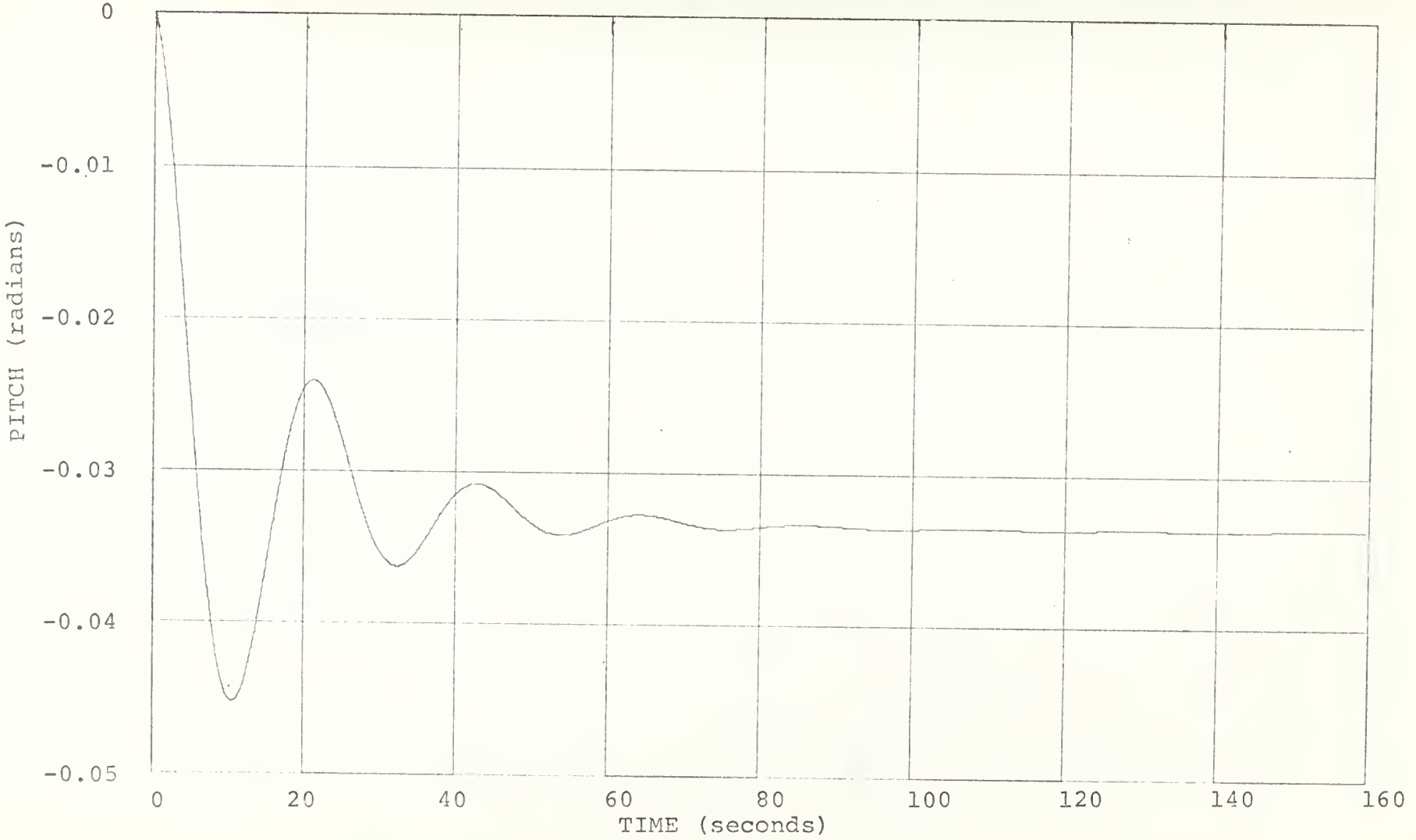


Figure 6-12 Submarine Pitch Response, $\delta s=0.1$, $u(0)=6$, $u_c=6$

VII. CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK

The technique developed in the present work is very general but directed to the solution of the equations of motion. The computer simulations involve usual maneuvers as the effects of deflection of rudders,canards or planes, but more realistic maneuvers can be studied by using subroutines or other functions different from the step function.

The equations of motion of surface ships at the present time do not include cross coupling terms and consequently the lateral and longitudinal dynamics have to be considered separately, but when the state of the art reaches the stage in which hydrodynamic coefficients are available, the computer program developed here can be used in all its capacity since it was derived as if these terms were present. For the time being they are given a value of zero.

Submarines have a more complete mathematical description and the set of equations given in Refs. 3 and 7 must be used. They include not only the nonlinear terms and cross coupling terms, but also some terms obtained empirically which are not readily obtainable from a standard derivation of the equations. The inclusion of all the terms in the computer program can be done when required.

The derivation of the method and the computer simulation were done with the idea of obtaining a program which can be used in control studies, then the use of compensators in feedback or cascade is an easy matter since all the variables

and some derivatives are available and the inputs to nodes are readily seen in the diagrams. Care must be taken in order to avoid the creation of implicit loops. Full use of the capabilities of the Continuous System Modeling Program (CSMP) is available for these types of studies.

Similarly more sophisticated studies can be done and the use of "MACROS", "PROCEDURE" or subprograms is prescribed to implement the simulation. The following implementations are suggested:

A. CONTROL OF THE RUDDER. To obtain a variable rate of deflection, with time delays.

B. BLOW OF TANKS. To control the blowing of the submarine's ballast tanks.

C. PROPELLER. To control the velocity of the ship by changing the rpm of the propeller.

COMPUTER PROGRAM 1 USING MEMORY

```
// EXEC CSMP360
//CSMP.SYSIN DD *
TITLE S3-LONG. DYNAMICS
PARAM U=1.
PARAM D1=0.1
PARAM UCZW=-4.303
PARAM UCZQ=-4.763
PARAM CZDE=-1.5
PARAM CZDC=-0.57
PARAM AWL=0.608
PARAM AXWL=-0.0647
PARAM ZW1=-1.075
PARAM ZQ1=-0.364
PARAM PY2=1.412
PARAM UCMW=-4.763
PARAM UCMQ=-15.08
PARAM CMDE=-2.85
PARAM CMDC=0.9
PARAM ZMP=1.045
PARAM MW1=-0.364
PARAM MQ1=-2.142
MEMORY BLOCK
MACRO  XOUT=BLOCK(XIN)
PROCEDURAL
       XOUT=XIN
ENDMAC
       AA=A*BAA+INTA*GAA
       AB=A*BAB+INTA*GAB
       BA=B*BBA+INTB*GBA
       BB=B*BBB+INTB*GBB
       AMEM=BLOCK(INA)
       ADOT=(1/AAA)*AMEM
       INA=I1-AA-BA-ABABD
       INB=I2-AB-BB-AABAD
       AABAD=AAB*ADOT
       ABABD=ABA*BDOT
       BDOT=(1/ABB)*INB
       A=INTGRL(0.,ADOT)
       INTA=INTGRL(0.,A)
       B=INTGRL(0.,BDOT)
       INTB=INTGRL(0.,B)
       I1=NL1+WIND1+WAVE1+IF1
       I2=NL2+WIND2+WAVE2+IF2
       IF1=KA1*D1+KA2*D2
       IF2=KB1*D1+KB2*D2
       KA1=-0.5*CZDC*U**2
       KB1=-0.5*CMDC*U**2
       AAA=ZW1-1.
       BAA=0.5*UCZW*U
       GAA=-AWL
       ABA=ZQ1
       BBA=ZW1*U+0.5*UCZQ*U
       GBA=AXWL+0.5*UCZW*U**2
       AAB=MW1
       BAB=0.5*UCMW*U
       GAB=AXWL
       ABB=MQ1-PY2
       BBB=MW1*U+0.5*UCMQ*U
       GBB=-ZMP+0.5*UCMW*U**2
PRINT INTA,INTB
PRTPLOT INTA
PRTPLOT INTB
TIMER FINTIM=60.,DELT=0.02,DELMIN=0.02,PRDEL=0.2
LABEL S3-LONG. DYNAMICS
END
STOP
ENDJOB
```

COMPUTER PROGRAM 2 USING IMPLICIT

```
// EXEC CSMP360
//CSMP.SYSIN DD *
TITLE S3-LONG. DYNAMICS
PARAM U=1.
PARAM D1=0.1
PARAM UCZW=-4.303
PARAM UCZQ=-4.763
PARAM CZDE=-1.5
PARAM CZDC=-0.57
PARAM AWL=0.608
PARAM AXWL=-0.0647
PARAM ZW1=-1.075
PARAM ZQ1=-0.364
PARAM PY2=1.412
PARAM UCMW=-4.763
PARAM UCMQ=-15.08
PARAM CMDE=-2.85
PARAM CMDC=0.9
PARAM ZMP=1.045
PARAM MW1=-0.364
PARAM MQ1=-2.142
      AA=A*BAA+INTA*GAA
      AB=A*BAB+INTA*GAB
      BA=B*BBA+INTB*GBA
      BB=B*BBB+INTB*GBB
      ADOT=IMPL(0.,0.1,FOFA)
      INA=I1-AA-BA-ABABD
      INB=I2-AB-BB-AABAD
      AABAD=AAB*ADOT
      ABABD=ABA*BDOT
      BDOT=(1/ABB)*INB
      FOFA=(1/AAA)*INA
      A=INTGRL(0.,ADOT)
      INTA=INTGRL(0.,A)
      B=INTGRL(0.,BDOT)
      INTB=INTGRL(0.,B)
      I1=NL1+WIND1+WAVE1+IF1
      I2=NL2+WIND2+WAVE2+IF2
      IF1=KA1*D1+KA2*D2
      IF2=KB1*D1+KB2*D2
      KA1=-0.5*CZDC*U**2
      KB1=-0.5*CMDC*U**2
      AAA=ZW1-1.
      BAA=0.5*UCZW*U
      GAA=-AWL
      ABA=ZQ1
      BBA=ZW1*U+0.5*UCZQ*U
      GBA=AXWL+0.5*UCZW*U**2
      AAB=MW1
      BAB=0.5*UCMW*U
      GAB=AXWL
      ABB=MQ1-PY2
      BBB=MW1*U+0.5*UCMQ*U
      GBB=-ZMP+0.5*UCMW*U**2
PRINT INTA,INTB
PRTPLOT INTA
PRTPLOT INTB
TIMER FINTIM=60.,DELT=0.02,DELMIN=0.02,PRDEL=0.2
LABEL S3-LONG. DYNAMICS
END
STOP
ENDJOB
```

COMPUTER PROGRAM 3
LATERAL DYNAMICS OF THE S3

```
// EXEC CSMP360
//CSMP.SYSIN DD *
*THIS PROGRAM SIMULATES THE DYNAMICS OF A SURFACE SHIP , THE EQUATIONS
* OF MOTION ARE ASSUMED TO BE IN THE FORM:
*(AAAS2+BAAS+GAA)A+(ABAS2+BBAS+GBA)B+(ACAS2+BCAS+GCA)C=KA1*D1+KA2*D2+NA
*(AABS2+BABS+GAB)A+(ABBS2+BBBS+GBB)B+(ACBS2+BCBS+GCB)C=KB1*D1+KB2*D2+NB
*(AACS2+BACS+GAC)A+(ABCS2+BBCS+GCB)B+(ACCS2+BCCS+GCC)C=KC1*D1+KC2*D2+NC
*A,B AND C ARE THE VARIABLES
*D1,D2 ARE THE DEFLECTIONS OF RUDDERS, CANARDS ETC.
*NA,NB,NC  ARE TERMS IN WHICH CAN BE INCLUDED THE COMPONENTS OF FORCES
*AND MOMENTS DUE TO WAVES AND WIND,SIMILARLY NON LINEAR TERMS
*WHICH CAN BE PRODUCTS,ABSOLUTE VALUES ETC. OF THE VARIABLES A,B,C AND
*THEIR FIRST DERIVATIVE,USING CSMP AND FORTRAN RULES
*AAB,BAB....,KA1...MUST BE DEFINED IN SECTION 2 ACCORDING TO THE
*HYDRODYNAMIC COEFFICIENTS,WHOSE VALUES ARE INTRODUCED IN SECTION 1,
*NO ORDER IS REQUIRED.
*IN SECTION 3 DEFINE VARIABLES ACCORDING TO PHYSICAL MEANING
*EG. YAW=A, THIS IS OPTIONAL
*VARIABLES A,B,C THEIR FIRST AND SECOND DERIVATIVE CAN BE PRINT-PLOTTED
*THE BASIC NAMES ARE:ADOT,BDOT,CDOT AND ADDOT,BDDOT,CDDOT
*IF THE REQUIRED SIMULATION INVOLVES ONLY ONE EQUATION DECLARE ABB AND ACC=1.
*THE SOLUTION WILL BE IN TERMS OF A.IF TWO EQUATIONS DECLARE ACC=1.,
*THE SOLUTION WILL BE IN TERMS OF A AND B.
*NON USED  TERMS ARE SET AUTOMATICALLY TO ZERO BY THE SORT SECTION
*OF THE SYSTEM, UP TO A MAXIMUM OF 85.THE EXCESS  MUST BE DECLARED IN
*SECTION 2
* SECTION 4 IS THE PROGRAMMED SIMULATION
*SECTION 5 , INTRODUCE DESIRED OUTPUT ,TIME,ACCORDING TO CSMP RULES
*
*SECTION 1 -COEFFICIENTS FOR A PARTICULAR SHIP
TITLE S3-LATERAL DYNAMICS
PARAM DR=0.1
PARAM U=1.
PARAM UCYV=-4.183
PARAM UCYP=1.668
PARAM UCYR=2.55
PARAM CYDR=-0.473
PARAM YV1=-1.886
PARAM YP1=0.964
PARAM YR1=0.208
PARAM PX2=1.086
PARAM UCLV=1.668
PARAM UCLP=-2.157
PARAM UCLR=-0.623
PARAM CLDR=0.351
```

```
PARAM CLDA=0.530
PARAM ZMR=0.0774
PARAM LV1=0.964
PARAM LPI=-1.99
PARAM LR1=-0.035
PARAM PZ2=2.478
PARAM UCNV=-3.860
PARAM UCNP=1.701
PARAM UCNR=-21.19
PARAM CNF=0.0735
PARAM CNDR=1.081
PARAM NV1=0.208
PARAM NP1=-0.035
PARAM NR1=-3.841
*SECTION 2 -PARAMETERS CALCULATIONS
      AAA=YV1-1.
      AAB=LV1
      AAC=NV1
      ABA=YR1
      ABB=LR1
      ABC=NR1-PZ2
      ACA=YP1
      ACB=LPI-PX2
      ACC=NP1
      BAA=0.5*UCYV*U
      BBA=0.5*UCYR*U-U
      BCA=0.5*UCYP*U
      BAB=0.5*UCLV*U
      BBB=0.5*UCLR*U
      BCB=0.5*UCLP*U
      GCB=-ZMR
      BAC=0.5*UCNV*U
      BBC=0.5*UCNR*U
      BCC=0.5*UCNP*U
      GCC=0.5*CNF*U**2
      KA1=-0.5*CYDR*U**2
      KA2=-0.5*CLDA*U**2
      KB1=-0.5*CLDR*U**2
      KB2=-0.5*CLDA*U**2
      KC1=-0.5*CNDR*U**2
*SECTION 3 -PHYSICAL DEFINITIONS
      YAW=B
      ROLL=C
      D1=DR
*SECTION 4 -PROGRAMMED SIMULATION
      ADDOT=(COFAA*I1+COFAB*I2+COFAC*I3)/DEL
      ADOT=INTGRL(0.,ADDOT)
      A=INTGRL(0.,ADOT)
```

```
      BDDOT=(COFBA*I1+COFBB*I2+COFBC*I3)/DEL
      BDOT=INTGRL(0.,BDDOT)
      B=INTGRL(0.,BDOT)
      CDDOT=(COFCA*I1+COFCB*I2+COFCC*I3)/DEL
      CDOT=INTGRL(0.,CDDOT)
      C=INTGRL(0.,CDOT)
      COFAA=ABB*ACC-ACB*ABC
      COFAB=-(ABA*ACC-ACA*ABC)
      COFAC=ABA*ACB-ABB*ACA
      COFBA=-(AAB*ACC-ACB*AAC)
      COFBB=AAA*ACC-ACA*AAC
      COFBC=-(AAA*ACB-ACA*AAB)
      COFCA=AAB*ABC-ABB*AAC
      COFCB=-(AAA*ABC-ABA*AAC)
      COFCC=AAA*ABB-ABA*AAB
      I1=-BAA*ADOT-GAA*A-BBA*BDOT-GBA*B-BCA*CDOT-GCA*C+IF1
      I2=-BAB*ADOT-GAB*A-BBB*BDOT-GBB*B-BCB*CDOT-GCB*C+IF2
      I3=-BAC*ADOT-GAC*A-BBC*BDOT-GBC*B-BCC*CDOT-GCC*C+IF3
      IF1=KA1*D1+KA2*D2+NA
      IF2=KB1*D1+KB2*D2+NB
      IF3=KC1*D1+KC2*D2+NC
      DEL=S1-S2
      S1=AAA*ABB*ACC+AAB*ABC*ACA+AAC*ACB*ABA
      S2=ACA*ABB*AAC+ACB*ABC*AAA+ACC*AAB*ABA
*SECTION 5 -OUTPUT CHARACTERISTICS
PRTPLOT ROLL
PRTPLOT YAW
TIMER  FINTIM=60.,DELT=0.02,PRDEL=0.2
LABEL S3-LATERAL DYNAMICS
END
STOP
ENDJOB
```

COMPUTER PROGRAM 4A
LATERAL AND LONGITUDINAL DYNAMICS OF THE S3

```
// EXEC CSMP360
//CSMP.SYSIN DD *
*THIS PROGRAM SIMULATES THE DYNAMICS OF A SHIP IN 6 DEGREES OF FREEDOM
*THE EQUATIONS OF MOTION ARE ASSUMED TO BE IN THE FORM:
* HAA*A+HBA*B+HCA*C+HDA*D+HEA*E+HFA*F=IFA
* HAB*A+HBB*B+HCB*C+HDB*D+HEB*E+HFB*F=IFB
* HAC*A+HBC*B+HCC*C+HDC*D+HEC*E+HFC*F=IFC
* HAD*A+HBD*B+HCD*C+HDD*D+HED*E+HFD*F=IFD
* HAE*A+HBE*B+HCE*C+HDE*D+HEE*E+HFE*F=IFE
* HAF*A+HBF*B+HCF*C+HDF*D+HEF*E+HFF*F=IFF
*HIJ ARE POLYNOMIALS OF THE FORM:AIJ*S**2+BIJ*S+GIJ
*WHERE I IS THE COLUMN AND J IS THE ROW
*A,B,C,D,E,F ARE THE VARIABLES
* IFJ=KJ1*D1+KJ2*D2+NJ
*AIJ,BIJ,GIJ,HJ1,KJ# MUST BE DEFINED IN SECTION 2 FROM THE HYDRODYNAMIC
*COEFFICIENTS AND THE NONDIMENSIONALIZATION PROCESS.
*THE VALUES FOR AIJ MUST BE DEFINED AS INDICATED IN SECTION 2A
*IF NOT ALL EQUATIONS ARE USED,THE NON USED AIJ MUST BE SET=1
*THIS SITUATION CAN ARISE WHEN ONLY THE LATERAL OR LONGITUDINAL DYN.IS DESIRED
*NON USED  TERMS ARE SET AUTOMATICALLY TO ZERO BY THE SORT SECTION
*OF THE SYSTEM, UP TO A MAXIMUM OF 85.THE EXCESS  MUST BE DECLARED IN
*SECTION 2
*D1,D2 ARE THE DEFLECTIONS OF RUDDERS,CANARDS ETC
*NJ ARE TERMS IN WHICH CAN BE INCLUDED THE COMPONENTS OF FORCES AND MOMENTS
*DUE TO WAVES AND/OR WIND.ALSO NON LINEAR TERMS CAN BE DEFINED BUT UP TO
*THE FIRST DERIVATIVE OF THE BASIC VARIABLES(A,B,C,D,E,F).IF OTHER  EFFECTS OR
*MORE RUDDERS ,CANARDS ETC. ARE REQUIRED THEY CAN ALSO BE INCLUDED IN IFJ
*ALL DEFINITIONS   CAN BE   INCLUDED IN SECTION 3.NO ORDER REQUIRED
*HYDRODYNAMICS COEFFICIENTS AND OTHER PARAMETERS ARE INTRODUCED IN SECTION 1
*
*THE VARIABLES A,B,.... THEIR FIRST DERIVATIVES ADOT,BDOT,.... AND
*THEIR SECOND DERIVATIVES ADDOT,BDDOT.... ARE AVAILABLE TO PRINT-PLOT
*
*THE DETERMINANT AND COFACTORS ARE PRINTED;COF00=DEL,COF11=COFAA,COF21=COFAB...
*
*THE PARAMETERS:M,M2,M1,K,L,N,LP1,KP,J,I DEFINED AS FIXED CAN NOT BE USED
*SINCE THEY ARE PART OF THE MAIN SIMULATION
* EXPLANATION OF THE THEORY FOR THE PRESENT PROGRAM CAN BE FOUND IN
* E.ROMERO THESIS JUN/72
FIXED M,M2,M1,K,L,N,LP1,KP,J,I
*SECTION 1 -COEFFICIENTS FOR A PARTICULAR SHIP
TITLE S3 LAT. AND LONG DYNAMICS
PARAM D1=0.1
PARAM U=1.
PARAM UCYV=-4.183
```

```
PARAM UCYP=1.668
PARAM UCYR=2.55
PARAM CYDR=-0.473
PARAM YV1=-1.886
PARAM YP1=0.964
PARAM YR1=0.208
PARAM PX2=1.086
PARAM UCLV=1.668
PARAM UCLP=-2.157
PARAM UCLR=-0.623
PARAM CLDR=0.351
PARAM CLDA=0.530
PARAM ZMR=0.0774
PARAM LV1=0.964
PARAM LP1=-1.99
PARAM LR1=-0.035
PARAM PZ2=2.478
PARAM UCNV=-3.860
PARAM UCNP=1.701
PARAM UCNR=-21.19
PARAM CNF=0.0735
PARAM CNDR=1.081
PARAM NV1=0.208
PARAM NP1=-0.035
PARAM NR1=-3.841
PARAM UCZW=-4.303
PARAM UCZQ=-4.763
PARAM CZDE=-1.5
PARAM CZDC=-0.57
PARAM AWL=0.608
PARAM AXWL=-0.0647
PARAM ZW1=-1.075
PARAM ZQ1=-0.364
PARAM PY2=1.412
PARAM UCMW=-4.763
PARAM UCMQ=-15.08
PARAM CMDE=-2.85
PARAM CMDC=0.9
PARAM ZMP=1.045
PARAM MW1=-0.364
PARAM MQ1=-2.142
*SECTION 2 -PARAMETERS CALCULATIONS
INITIAL
*SECTION 2A -ALL PARAMETERS MUST BE DEFINED AND IN SEQUENCE:
*AAA,AAB,AAC,AAD,AAE,AAF,ABA,ABB,ABC,ABD,ABE,ABF,ACA,ACB,ACC,ACD,ACE
*ACF,ADA,ADB,ADC,ADD,ADE,ADF,AEA,AEB,AEC,AED,AEE,AEF,AFA,AFB,AFC,AFD,AFE,AFF
      AAA=YV1-1.
      AAB=LV1
```

```
      AAC=NV1
      AAD=0.
      AAE=0.
      AAF=0.
      ABA=YR1
      ABB=LR1
      ABC=NR1-PZ2
      ABD=0.
      ABE=0.
      ABF=0.
      ACA=YP1
      ACB=LPI-PX2
      ACC=NP1
      ACD=0.
      ACE=0.
      ACF=0.
      ADA=0.
      ADB=0.
      ADC=0.
      ADD=1.
      ADE=0.
      ADF=0.
      AEA=0.
      AEB=0.
      AEC=0.
      AED=0.
      AEE=ZW1-1.
      AEF=MW1
      AFA=0.
      AFB=0.
      AFC=0.
      AFD=0.
      AFE=ZQ1
      AFF=MQ1-PY2
*END SECTION 2A
      DEL=VALUE(AAA,0,0)
      COFAA=VALUE(AAA,1,1)
      COFAB=VALUE(AAA,2,1)
      COFAC=VALUE(AAA,3,1)
      COFAD=VALUE(AAA,4,1)
      COFAE=VALUE(AAA,5,1)
      COFAF=VALUE(AAA,6,1)
      COFBA=VALUE(AAA,1,2)
      COFBB=VALUE(AAA,2,2)
      COFBC=VALUE(AAA,3,2)
      COFBD=VALUE(AAA,4,2)
      COFBE=VALUE(AAA,5,2)
      COFBF=VALUE(AAA,6,2)
```

```
      COFCA=VALUE(AAA,1,3)
      COFCB=VALUE(AAA,2,3)
      COFCC=VALUE(AAA,3,3)
      COFCD=VALUE(AAA,4,3)
      COFCE=VALUE(AAA,5,3)
      COFCF=VALUE(AAA,6,3)
      COFDA=VALUE(AAA,1,4)
      COFDB=VALUE(AAA,2,4)
      COFDC=VALUE(AAA,3,4)
      COFDD=VALUE(AAA,4,4)
      COFDE=VALUE(AAA,5,4)
      COFDF=VALUE(AAA,6,4)
      COFEA=VALUE(AAA,1,5)
      COFEB=VALUE(AAA,2,5)
      COFEC=VALUE(AAA,3,5)
      COFED=VALUE(AAA,4,5)
      COFEE=VALUE(AAA,5,5)
      COFEF=VALUE(AAA,6,5)
      COFFA=VALUE(AAA,1,6)
      COFFB=VALUE(AAA,2,6)
      COFFC=VALUE(AAA,3,6)
      COFFD=VALUE(AAA,4,6)
      COFFE=VALUE(AAA,5,6)
      COFFF=VALUE(AAA,6,6)
DYNAMIC
      BAA=0.5*UCYV*U
      BBA=0.5*UCYR*U-U
      BCA=0.5*UCYP*U
      BAB=0.5*UCLV*U
      BBB=0.5*UCLR*U
      BCB=0.5*UCLP*U
      GCB=-ZMR
      BAC=0.5*UCNV*U
      BBC=0.5*UCNR*U
      BCC=0.5*UCNP*U
      GCC=0.5*CNF*U**2
      BEE=0.5*UCZW*U
      GEE=-AWL
      BFE=ZW1*U+0.5*UCZQ*U
      GFE=AXWL+0.5*UCZW*U**2
      BEF=0.5*UCMW*U
      GEF=AXWL
      BFF=MW1*U+0.5*UCMQ*U
      GFF=-ZMP+0.5*UCMW*U**2
      KA1=-0.5*CYDR*U**2
      KA2=-0.5*CLDA*U**2
      KB1=-0.5*CLDR*U**2
      KB2=-0.5*CLDA*U**2
```

```
      KC1=-0.5*CNDR*U**2
      KE1=-0.5*CZDC*U**2
      KF1=-0.5*CMDC*U**2
*SECTION 3-DEFINITIONS
      YAW=B
      ROLL=C
      HEAVE=E
      PITCH=F
*SECTION 4 -PROGRAMMED SIMULATION
      I1=-BAA*ADOT-GAA*A-BBA*BDOT-GBA*B-BCA*CDOT-GCA*C...
      -BDA*DDOT-GDA*D-BEA*EDOT-GEA*E-BFA*FDOT-GFA*F+IF1
      I2=-BAB*ADOT-GAB*A-BBB*BDOT-GBB*B-BCB*CDOT-GCB*C...
      -BDB*DDOT-GDB*D-BEB*EDOT-GEB*E-BFB*FDOT-GFB*F+IF2
      I3=-BAC*ADOT-GAC*A-BBC*BDOT-GBC*B-BCC*CDOT-GCC*C...
      -BDC*DDOT-GDC*D-BEC*EDOT-GEC*E-BFC*FDOT-GFC*F+IF3
      I4=-BAD*ADOT-GAD*A-BBD*BDOT-GBD*B-BCD*CDOT-GCD*C...
      -BDD*DDOT-GDD*D-BED*EDOT-GFD*E-BFD*FDOT-GFD*F+IF4
      I5=-BAE*ADOT-GAE*A-BBE*BDOT-GBE*B-BCE*CDOT-GCE*C...
      -BDE*DDOT-GDE*D-BEE*EDOT-GEE*E-BFE*FDOT-GFE*F+IF5
      I6=-BAF*ADOT-GAF*A-BBF*BDOT-GBF*B-BCF*CDOT-GCF*C...
      -BDF*DDOT-GDF*D-BEF*EDOT-GEF*E-BFF*FDOT-GFF*F+IF6
      IF1=KA1*D1+KA2*D2+NA
      IF2=KB1*D1+KB2*D2+NB
      IF3=KC1*D1+KC2*D2+NC
      IF4=KD1*D1+KD2*D2+ND
      IF5=KE1*D1+KE2*D2+NE
      IF6=KF1*D1+KF2*D2+NF
      ADDOT=(COFAA*I1+COFAB*I2+COFAC*I3+COFAD*I4+COFAE*I5+COFAF*I6)/DEL
      BDDOT=(COFBA*I1+COFBB*I2+COFBC*I3+COFBD*I4+COFBE*I5+COFBF*I6)/DEL
      CDDOT=(COFCA*I1+COFCB*I2+COFCC*I3+COFCD*I4+COFCE*I5+COFCF*I6)/DEL
      DDDOT=(COFDA*I1+COFDB*I2+COFDC*I3+COFDD*I4+COFDE*I5+COFDF*I6)/DEL
      EDDOT=(COFEA*I1+COFEB*I2+COFEC*I3+COFED*I4+COFEE*I5+COFEF*I6)/DEL
      FDDOT=(COFFA*I1+COFFB*I2+COFFC*I3+COFFD*I4+COFFE*I5+COFFF*I6)/DEL
      ADOT=INTGRL(ADOTO,ADDOT)
      BDOT=INTGRL(0.,BDDOT)
      CDOT=INTGRL(0.,CDDOT)
      DDOT=INTGRL(0.,DDDOT)
      EDOT=INTGRL(0.,EDDOT)
      FDOT=INTGRL(0.,FDDOT)
      A=INTGRL(0.,ADOT)
      B=INTGRL(0.,BDOT)
      C=INTGRL(0.,CDOT)
      D=INTGRL(0.,DDOT)
      E=INTGRL(0.,EDOT)
      F=INTGRL(0.,FDOT)
*SECTION 5 -OUTPUT CHARACTERISTICS
PRINT YAW,ROLL,HEAVE,PITCH
PRTPLOT YAW
```

```
PRTPLOT ROLL
PRTPLOT HEAVE
PRTPLOT PITCH
TIMER  FINTIM=50.,DELT=0.02,PRDEL=0.2
LABEL S3 LAT. AND LONG. DYNAMICS
END
STOP


      FUNCTION VALUE(Y,I,M)
COMMON
      DIMENSION X(6,6),Y(6,6)
      DO 1 M1=1,6
      DO 1 M2=1,6
    1  X(M1,M2)=Y(M1,M2)
      IF(I.EQ.0) GO TO 100
      X(I,1)=0.
      X(I,2)=0.
      X(I,3)=0.
      X(I,4)=0.
      X(I,5)=0.
      X(I,6)=0.
      X(1,M)=0.
      X(2,M)=0.
      X(3,M)=0.
      X(4,M)=0.
      X(5,M)=0.
      X(6,M)=0.
      X(I,M)=1.
  100 CONTINUE
      WRITE(6,49) I,M
   49 FORMAT(/,' ','DETERMINANT FOR COF',I1,I1,':')
      DO 50 M1=1,6
      WRITE(6,51)(X(M1,M2),M2=1,6)
   51 FORMAT(10(1X,E13.6))
   50 CONTINUE
      N=6
      DD=1.DO
   10  DO 34 L=1,N
       KP=0
       Z=0.0
       DO 12 K=L,N
      IF(Z-ABS(X(K,L)))11,12,12
   11 Z=ABS(X(K,L))
       KP=K
   12  CONTINUE
       IF(L-KP)13,20,20
   13  DO 14 J=L,N
```

```
          Z=X(L,J)
          X(L,J)=X(KP,J)
   14     X(KP,J)=Z
          DD=-DD
   20     IF(L-N)31,40,40
   31     LP1=L+1
          DO 34 K=LP1,N
          IF(X(K,L))32,34,32
   32     RATIO=X(K,L)/X(L,L)
          DO 33 J=LP1,N
   33     X(K,J)=X(K,J)-RATIO*X(L,J)
   34     CONTINUE
   40   DO 41 K=1,N
   41   DD=DD*X(K,K)
          D=DD
          VALUE=D
          WRITE(6,52) I,M,VALUE
   52   FORMAT(' ','COF',I1,I1,'=',E15.6)
          RETURN
          END
ENDJOB
```

COMPUTER PROGRAM 4B,TRANSFER FUNCTION

```
//JOBLIB      DD    DISP=SHR,DSNAME=SYS1.DSL
//         EXEC DSL1,REGION.G=130K,TIME.G=2
//DSL.INPUT   DD    *          *** DSL PROBLEM FOLLOWS THIS CARD ***
TITLE TRANSFER FUNCTION
PARAM IN=0.1
      P=TRNFR(4,6,IC,NUM,DEN,IN)
STORAG IC(6),NUM(5),DEN(7)
TABLE IC(1-6)=6*0,NUM(1-5)=4.2098,5.6160,1.3335,0.12987,0.,...
      DEN(1-7)=50.1011,135.2824,97.2277,20.3955,1.772,0.,0.
CONTRL DELT=.02,FINTIM=60
PRINT .2,P
PREPAR .1,P
GRAPH TIME,P
LABEL CHECK
PRPLOT ONLY
END
STOP
```

COMPUTER PROGRAM 5
SUBMARINE SIX DEGREES OF FREEDOM

```
// EXEC CSMP360
//CSMP.SYSIN DD *
*THIS PROGRAM SIMULATES THE DYNAMICS OF A SHIP IN 6 DEGREES OF FREEDOM
*THE EQUATIONS OF MOTION ARE ASSUMED TO BE IN THE FORM:
* HAA*A+HBA*B+HCA*C+HDA*D+HEA*E+HFA*F=IFA
* HAB*A+HBB*B+HCB*C+HDB*D+HEB*E+HFB*F=IFB
* HAC*A+HBC*B+HCC*C+HDC*D+HEC*E+HFC*F=IFC
* HAD*A+HBD*B+HCD*C+HDD*D+HED*E+HFD*F=IFD
* HAE*A+HBE*B+HCE*C+HDE*D+HEE*E+HFE*F=IFE
* HAF*A+HBF*B+HCF*C+HDF*D+HEF*E+HFF*F=IFF
*HIJ ARE POLYNOMIALS OF THE FORM:AIJ*S**2+BIJ*S+GIJ
*WHERE I IS THE COLUMN AND J IS THE ROW
*A,B,C,D,E,F ARE THE VARIABLES
* IFJ=KJ1*D1+KJ2*D2+NJ
*AIJ,BIJ,GIJ,HJ1,KJ# MUST BE DEFINED IN SECTION 2 FROM THE HYDRODYNAMIC
*COEFFICIENTS AND THE NONDIMENSIONALIZATION PROCESS.
*THE VALUES FOR AIJ MUST BE DEFINED AS INDICATED IN SECTION 2A
*IF NOT ALL EQUATIONS ARE USED,THE NON USED AIJ MUST BE SET=1
*THIS SITUATION CAN ARISE WHEN ONLY THE LATERAL OR LONGITUDINAL DYN.IS DESIRED
*NON USED  TERMS ARE SET AUTOMATICALLY TO ZERO BY THE SORT SECTION
*OF THE SYSTEM, UP TO A MAXIMUM OF 85.THE EXCESS  MUST BE DECLARED IN
*SECTION 2
*D1,D2 ARE THE DEFLECTIONS OF RUDDERS,CANARDS ETC
*NJ ARE TERMS IN WHICH CAN BE INCLUDED THE COMPONENTS OF FORCES AND MOMENTS
*DUE TO WAVES AND/OR WIND.ALSO NON LINEAR TERMS CAN BE DEFINED BUT UP TO
*THE FIRST DERIVATIVE OF THE BASIC VARIABLES(A,B,C,D,E,F).IF OTHER  EFFECTS OR
*MORE RUDDERS ,CANARDS ETC. ARE REQUIRED THEY CAN ALSO BE INCLUDED IN IFJ
*ALL DEFINITIONS   CAN BE   INCLUDED IN SECTION 3.NO ORDER REQUIRED
*HYDRODYNAMICS COEFFICIENTS AND OTHER PARAMETERS ARE INTRODUCED IN SECTION 1
*
*THE VARIABLES A,B,.... THEIR FIRST DERIVATIVES ADOT,BDOT,.... AND
*THEIR SECOND DERIVATIVES ADDOT,BDDOT.... ARE AVAILABLE TO PRINT-PLOT
*
*THE DETERMINANT AND COFACTORS ARE PRINTED;COF00=DEL,COF11=COFAA,COF21=COFAB...
*
*THE PARAMETERS:M,M2,M1,K,L,N,LP1,KP,J,I DEFINED AS FIXED CAN NOT BE USED
*SINCE THEY ARE PART OF THE MAIN SIMULATION
* EXPLANATION OF THE THEORY FOR THE PRESENT PROGRAM CAN BE FOUND IN
* E.ROMERO THESIS JUN/72
FIXED M,M2,M1,K,L,N,LP1,KP,J,I
*SECTION 1 -COEFFICIENTS FOR A PARTICULAR SHIP
TITLE SUBMARINE SIX DEGREES OF FREEDOM
PARAM DR=-0.1
PARAM LC=20.
PARAM UC=6.
```

```
INCON ADOT0=6.
PARAM ML=0.259
PARAM A1=-0.001766
PARAM A2=-0.001403
PARAM A3=0.003169
PARAM IX=0.0004354
PARAM IY=0.0134638
PARAM IZ=0.0134638
PARAM BDZVP=0.01047
PARAM BDMRP=-0.000944
PARAM XUDOT=-0.000484
PARAM XVR=0.01818
PARAM XWQ=-0.01474
PARAM XVV=0.01733
PARAM XDRDR=-0.00512
PARAM XDSDS=-0.00288
PARAM XDBDB=-0.00347
PARAM XQQ=-0.000517
PARAM XRR=0.000185
PARAM XRP=0.0003885
PARAM YVDOT=-0.01818
PARAM YWP=0.01474
PARAM YV=-0.05547
PARAM Y1V1V=-0.09280
PARAM YR=0.00138
PARAM YV1R1=-0.0119
PARAM YP=-0.00213
PARAM YRDOT=-0.000185
PARAM YPDOT=-0.0003885
PARAM YDR=0.01218
PARAM YPQ=0.000517
PARAM YWV=-0.1297
PARAM ZWDOT=-0.01474
PARAM ZVP=-0.01818
PARAM ZS=-0.00001
PARAM ZW=-0.02342
PARAM ZW1W1=-0.06
PARAM ZVV=0.1297
PARAM ZQ=-0.00962
PARAM ZW1Q1=-0.0119
PARAM ZVR=-0.01951
PARAM ZRR=-0.002617
PARAM ZDS=-0.01134
PARAM ZDB=-0.00558
PARAM ZQDOT=-0.000517
PARAM Z1W1=-0.001253
PARAM ZWW=0.0004514
PARAM ZRP=-0.002471
```

```
PARAM KPDOT=-0.0000313
PARAM KQR=0.000036
PARAM KRDOT=-0.0000358
PARAM K1P1P=-0.000017
PARAM KV=-0.004454
PARAM K1V1V=-0.003998
PARAM KPC=-0.000162
PARAM KR=-0.0007317
PARAM KVDOT=-0.000315
PARAM KVW=0.00291
PARAM KDR=0.0000496
PARAM KWP=0.0003885
PARAM MQDOT=-0.000787
PARAM MRP=0.00072
PARAM MS=0.0001418
PARAM MW=0.004757
PARAM M1W1W=-0.01104
PARAM MVV=0.03061
PARAM MQ=-0.0055
PARAM M1W1Q=-0.00553
PARAM MVR=-0.009073
PARAM MRR=-0.001
PARAM MWDOT=-0.000584
PARAM MDS=-0.00532
PARAM MDB=0.001137
PARAM M1W1=-0.0007886
PARAM MWW=0.002095
PARAM MVP=0.002471
PARAM NRDOT=-0.000751
PARAM NPQ=-0.000787
PARAM NPDOT=-0.0000358
PARAM NV=-0.014830
PARAM N1V1V=0.01635
PARAM NR=-0.004917
PARAM N1V1R=-0.00553
PARAM NP=-0.000377
PARAM NVDOT=-0.000141
PARAM NDR=-0.00567
PARAM NWV=0.03061
PARAM NWP=-0.000517
*SECTION 2 -PARAMETERS CALCULATIONS
INITIAL
*SECTION 2A -ALL PARAMETERS MUST BE DEFINED AND IN SEQUENCE:
*AAA,AAB,AAC,AAD,AAE,AAF,ABA,ABB,ABC,ABD,ABE,ABF,ACA,ACB,ACC,ACD,ACE
*ACF,ADA,ADB,ADC,ADD,ADE,ADF,AEA,AEB,AEC,AED,AEE,AEF,AFA,AFB,AFC,AFD,AFE,AFF
      AAA=XUDOT-ML
      AAB=0.
      AAC=0.
```

```
      AAD=0.
      AAE=0.
      AAF=0.
      ABA=0.
      ABB=YVDOT-ML
      ABC=0.
      ABD=KVDOT/LC
      ABE=0.
      ABF=NVDOT/LC
      ACA=0.
      ACB=0.
      ACC=ZWDOT-ML
      ACD=0.
      ACE=MWDOT/LC
      ACF=0.
      ADA=0.
      ADB=LC*YPDOT
      ADC=0.
      ADD=KPDOT-IX
      ADE=0.
      ADF=NPDOT
      AEA=0.
      AEB=0.
      AEC=LC*ZQDOT
      AED=0.
      AEE=MQDOT-IY
      AEF=0.
      AFA=0.
      AFB=LC*YRDOT
      AFC=0.
      AFD=KRDOT
      AFE=0.
      AFF=NRDOT-IZ
*END SECTION 2A
      DEL=VALUE(AAA,0,0)
      COFAA=VALUE(AAA,1,1)
      COFAB=VALUE(AAA,2,1)
      COFAC=VALUE(AAA,3,1)
      COFAD=VALUE(AAA,4,1)
      COFAE=VALUE(AAA,5,1)
      COFAF=VALUE(AAA,6,1)
      COFBA=VALUE(AAA,1,2)
      COFBB=VALUE(AAA,2,2)
      COFBC=VALUE(AAA,3,2)
      COFBD=VALUE(AAA,4,2)
      COFBE=VALUE(AAA,5,2)
      COFBF=VALUE(AAA,6,2)
      COFCA=VALUE(AAA,1,3)
```

```
      COFCB=VALUE(AAA,2,3)
      COFCC=VALUE(AAA,3,3)
      COFCD=VALUE(AAA,4,3)
      COFCE=VALUE(AAA,5,3)
      COFCF=VALUE(AAA,6,3)
      COFDA=VALUE(AAA,1,4)
      COFDB=VALUE(AAA,2,4)
      COFDC=VALUE(AAA,3,4)
      COFDD=VALUE(AAA,4,4)
      COFDE=VALUE(AAA,5,4)
      COFDF=VALUE(AAA,6,4)
      COFEA=VALUE(AAA,1,5)
      COFEB=VALUE(AAA,2,5)
      COFEC=VALUE(AAA,3,5)
      COFED=VALUE(AAA,4,5)
      COFEE=VALUE(AAA,5,5)
      COFEF=VALUE(AAA,6,5)
      COFFA=VALUE(AAA,1,6)
      COFFB=VALUE(AAA,2,6)
      COFFC=VALUE(AAA,3,6)
      COFFD=VALUE(AAA,4,6)
      COFFE=VALUE(AAA,5,6)
      COFFF=VALUE(AAA,6,6)
DYNAMIC
      BBB=U*YV/LC
      BBD=U*KV/LC**2
      BBF=U*NV/LC**2
      BCC=U*ZW/LC
      BCE=U*MW/LC**2
      BDB=U*YP
      BDD=U*KPC/LC
      BDF=U*NP/LC
      BEC=U*ZQ
      BEE=U*MQ/LC
      BFB=U*YR
      BFD=U*KR/LC
      BFF=U*NR/LC
      KA1=-XDRDR*U*U*DR/LC
      KA2=-XDSDS*U*U*DS/LC
      KA3=-XDBDB*U*U*DB/LC
      KB1=-YDR*U*U/LC
      KC2=-ZDS*U*U/LC
      KC3=-ZDB*U*U/LC
      KD1=-KDR*U*U/LC**2
      KE2=-MDS*U*U/LC**2
      KE3=-MDB*U*U/LC**2
      KF1=-NDR*U*U/LC**2
*SECTION 3-DEFINITIONS
```

```
      UDOT=ADDOT
      U=ADOT
      VDOT=BDDOT
      V=BDOT
      WDOT=CDDOT
      W=CDOT
      PDOT=DDDOT
      P=DDOT
      QDOT=EDDOT
      Q=EDOT
      RDOT=FDDOT
      R=FDOT
      D1=DP
      D2=DS
      D3=DB
*NON LINEAR RELATIONS
      NA=NA1+NA2+NA3+NA4
      NB=NB1+NB2+NB3
      NC=NC1+NC2+NC3
      ND=ND1+ND2+ND3
      NE=NE1+NE2+NE3
      NF=NF1+NF2+NF3
      NA1=-LC*(XQQ*Q**2+XRR*R**2+XRP*R*P)
      NA2=-(ML*V*R+XVR*V*R+XWQ*W*Q-ML*W*Q)
      NA3=-(XVV*V**2)/LC
      NA4=-(A1*U**2+A2*U*UC+A3*UC**2)/LC
      NB1=-LC*YPQ*P*Q
      NB2=-(YWP*W*P+YV1R1*V*ABR-ML*U*R+ML*W*P)
      NB3=-(YWV*W*V+Y1V1V*ABV*V)/LC
      NC1=-LC*(ZRR*R**2+ZRP*R*P)
      NC2=-(ZVP*V*P+ZVR*V*R+ZW1Q1*W*ABQ-ML*P*V)-BDZVP*V*P-ML*U*Q
      NC3=-(ZWW*W**2+ZVV*V**2+ZW1W1*W*ABW+U*Z1W1*ABW+U*U*ZS)/LC
      ND1=-(KQR*Q*R+K1P1P*ABP*P)+(IZ-IY)*Q*R
      ND2=-KWP*W*P/LC+(.518*0.0635*SIN(ROLL)*COS(PITCH))/LC
      ND3=-(K1V1V*ABV*V+KVW*V*W+K1W1V*ABW*V+KS)/LC**2
      NE1=-(MPP*R*P+MRR*R**2)-BDMRP*R*P+(IX-IZ)*R*P
      NE2=-(MVR*V*R+MVP*V*P+M1W1Q*ABW*Q)/LC+(.518*0.0635*SIN(PITCH))/LC
      NE3=-(MVV*V**2+MWW*W**2+M1W1W*ABW*W+U*M1W1*ABW+U*U*MS)/LC**2
      NF1=-NPQ*P*Q+(IY-IX)*P*Q
      NF2=-(NWP*W*P+N1V1R*ABV*R)/LC
      NF3=-(NWV*W*V+N1V1V*ABV*V)/LC**2
    & ABR=ABS(R)
    ? ABV=ABS(V)
    ? ABQ=ABS(Q)
    ? ABW=ABS(W)
    ? ABP=ABS(P)
*KINEMATIC RELATIONS
      RODOT=P+YADOT*SIN(PITCH)
```

```
      PIDOT=Q*COS(ROLL)-R*SIN(ROLL)
      YADOT=(R+PIDOT*SIN(ROLL))/COS(PITCH)*COS(ROLL)
      ROLL=INTGRL(0.,RODOT)
      PITCH=INTGRL(0.,PIDOT)
      YAW=INTGRL(0.,YADOT)
*SECTION 4 -PROGRAMMED SIMULATION
      I1=-BAA*ADOT-GAA*A-BBA*BDOT-GBA*B-BCA*CDOT-GCA*C...
      -BDA*DDOT-GDA*D-BEA*EDOT-GEA*E-BFA*FDOT-GFA*F+IF1
      I2=-BAB*ADOT-GAB*A-BBB*BDOT-GBB*B-BCB*CDOT-GCB*C...
      -BDB*DDOT-GDB*D-BEB*EDOT-GEB*E-BFB*FDOT-GFB*F+IF2
      I3=-BAC*ADOT-GAC*A-BBC*BDOT-GBC*B-BCC*CDOT-GCC*C...
      -BDC*DDOT-GDC*D-BEC*EDOT-GEC*E-BFC*FDOT-GFC*F+IF3
      I4=-BAD*ADOT-GAD*A-BBD*BDOT-GBD*B-BCD*CDOT-GCD*C...
      -BDD*DDOT-GDD*D-BED*EDOT-GED*E-BFD*FDOT-GFD*F+IF4
      I5=-BAE*ADOT-GAE*A-BBE*BDOT-GBE*B-BCE*CDOT-GCE*C...
      -BDE*DDOT-GDE*D-BEE*EDOT-GEE*E-BFE*FDOT-GFE*F+IF5
      I6=-BAF*ADOT-GAF*A-BBF*BDOT-GBF*B-BCF*CDOT-GCF*C...
      -BDF*DDOT-GDF*D-BEF*EDOT-GEF*E-BFF*FDOT-GFF*F+IF6
      IF1=KA1*D1+KA2*D2+KA3*D3+NA
      IF2=KB1*D1+KB2*D2+KB3*D3+NB
      IF3=KC1*D1+KC2*D2+KC3*D3+NC
      IF4=KD1*D1+KD2*D2+KD3*D3+ND
      IF5=KE1*D1+KE2*D2+KE3*D3+NE
      IF6=KF1*D1+KF2*D2+KF3*D3+NF
      ADDOT=(COFAA*I1+COFAB*I2+COFAC*I3+COFAD*I4+COFAE*I5+COFAF*I6)/DEL
      BDDOT=(COFBA*I1+COFBB*I2+COFBC*I3+COFBD*I4+COFBE*I5+COFBF*I6)/DEL
      CDDOT=(COFCA*I1+COFCB*I2+COFCC*I3+COFCD*I4+COFCE*I5+COFCF*I6)/DEL
      DDDOT=(COFDA*I1+COFDB*I2+COFDC*I3+COFDD*I4+COFDE*I5+COFDF*I6)/DEL
      EDDOT=(COFEA*I1+COFEB*I2+COFEC*I3+COFED*I4+COFEE*I5+COFEF*I6)/DEL
      FDDOT=(COFFA*I1+COFFB*I2+COFFC*I3+COFFD*I4+COFFE*I5+COFFF*I6)/DEL
      ADOT=INTGRL(ADOTO,ADDOT)
      BDOT=INTGRL(0.,BDDOT)
      CDOT=INTGRL(0.,CDDOT)
      DDOT=INTGRL(0.,DDDOT)
      EDOT=INTGRL(0.,EDDOT)
      FDOT=INTGRL(0.,FDDOT)
      A=INTGRL(0.,ADOT)
      B=INTGRL(0.,BDOT)
      C=INTGRL(0.,CDOT)
      D=INTGRL(0.,DDOT)
      E=INTGRL(0.,EDOT)
      F=INTGRL(0.,FDOT)
*SECTION 5 -OUTPUT CHARACTERISTICS
PREPAR TIME,YAW
TIMER FINTIM=160.,DELT=0.02,PRDEL=0.2
LABEL SUBMARINE SIX DEGREES OF FREEDOM
END
STOP
```

```
      FUNCTION VALUE(Y,I,M)
COMMON
      DIMENSION X(6,6),Y(6,6)
      DO 1 M1=1,6
      DO 1 M2=1,6
    1  X(M1,M2)=Y(M1,M2)
      IF(I.EQ.0) GO TO 100
      X(I,1)=0.
      X(I,2)=0.
      X(I,3)=0.
      X(I,4)=0.
      X(I,5)=0.
      X(I,6)=0.
      X(1,M)=0.
      X(2,M)=0.
      X(3,M)=0.
      X(4,M)=0.
      X(5,M)=0.
      X(6,M)=0.
      X(I,M)=1.
  100 CONTINUE
      WRITE(6,49) I,M
   49 FORMAT(/,' ','DETERMINANT FOR COF',I1,I1,':')
      DO 50 M1=1,6
      WRITE(6,51)(X(M1,M2),M2=1,6)
   51 FORMAT(10(1X,E13.6))
   50 CONTINUE
      N=6
      DD=1.DO
   10  DO 34 L=1,N
       KP=0
       Z=0.0
       DO 12 K=L,N
      IF(Z-ABS(X(K,L)))11,12,12
   11 Z=ABS(X(K,L))
       KP=K
   12  CONTINUE
       IF(L-KP)13,20,20
   13  DO 14 J=L,N
       Z=X(L,J)
       X(L,J)=X(KP,J)
   14  X(KP,J)=Z
      DD=-DD
   20  IF(L-N)31,40,40
   31  LP1=L+1
       DO 34 K=LP1,N
```

```
      IF(X(K,L))32,34,32
   32 RATIO=X(K,L)/X(L,L)
      DO 33 J=LP1,N
   33 X(K,J)=X(K,J)-RATIO*X(L,J)
   34 CONTINUE
   40 DO 41 K=1,N
   41 DD=DD*X(K,K)
      D=DD
      VALUE=D
      WRITE(6,52) I,M,VALUE
   52 FORMAT(' ','COF',I1,I1,'=',E15.6)
      RETURN
      END
ENDJOB
```

COMPUTER PROGRAM 6
PROGRAM EXPLANATION WITHOUT DATA

```
// EXEC CSMP360
//CSMP.SYSIN DD *
*THIS PROGRAM SIMULATES THE DYNAMICS OF A SHIP IN 6 DEGREES OF FREEDOM
*THE EQUATIONS OF MOTION ARE ASSUMED TO BE IN THE FORM:
* HAA*A+HBA*B+HCA*C+HDA*D+HEA*E+HFA*F=IFA
* HAB*A+HBB*B+HCB*C+HDB*D+HEB*E+HFB*F=IFB
* HAC*A+HBC*B+HCC*C+HDC*D+HEC*E+HFC*F=IFC
* HAD*A+HBD*B+HCD*C+HDD*D+HED*E+HFD*F=IFD
* HAE*A+HBE*B+HCE*C+HDE*D+HEE*E+HFE*F=IFE
* HAF*A+HBF*B+HCF*C+HDF*D+HEF*E+HFF*F=IFF
*HIJ ARE POLYNOMIALS OF THE FORM:AIJ*S**2+BIJ*S+GIJ
*WHERE I IS THE COLUMN AND J IS THE ROW
*A,B,C,D,E,F ARE THE VARIABLES
* IFJ=KJ1*D1+KJ2*D2+NJ
*AIJ,BIJ,GIJ,HJ1,KJ# MUST BE DEFINED IN SECTION 2 FROM THE HYDRODYNAMIC
*COEFFICIENTS AND THE NONDIMENSIONALIZATION PROCESS.
*THE VALUES FOR AIJ MUST BE DEFINED AS INDICATED IN SECTION 2A
*IF NOT ALL EQUATIONS ARE USED,THE NON USED AIJ MUST BE SET=1
*THIS SITUATION CAN ARISE WHEN ONLY THE LATERAL OR LONGITUDINAL DYN.IS DESIRED
*NON USED  TERMS ARE SET AUTOMATICALLY TO ZERO BY THE SORT SECTION
*OF THE SYSTEM, UP TO A MAXIMUM OF 85.THE EXCESS  MUST BE DECLARED IN
*SECTION 2
*D1,D2 ARE THE DEFLECTIONS OF RUDDERS,CANARDS ETC
*NJ ARE TERMS IN WHICH CAN BE INCLUDED THE COMPONENTS OF FORCES AND MOMENTS
*DUE TO WAVES AND/OR WIND.ALSO NON LINEAR TERMS CAN BE DEFINED BUT UP TO
*THE FIRST DERIVATIVE OF THE BASIC VARIABLES(A,B,C,D,E,F).IF OTHER  EFFECTS OR
*MORE RUDDERS ,CANARDS ETC. ARE REQUIRED THEY CAN ALSO BE INCLUDED IN IFJ
*ALL DEFINITIONS  CAN BE  INCLUDED IN SECTION 3.NO ORDER REQUIRED
*HYDRODYNAMICS COEFFICIENTS AND OTHER PARAMETERS ARE INTRODUCED IN SECTION 1
*
*THE VARIABLES A,B,.... THEIR FIRST DERIVATIVES ADOT,BDOT,.... AND
*THEIR SECOND DERIVATIVES ADDOT,BDDOT.... ARE AVAILABLE TO PRINT-PLOT
*
*THE DETERMINANT AND COFACTORS ARE PRINTED;COF00=DEL,COF11=COFAA,COF21=COFAB...
*
*THE PARAMETERS:M,M2,M1,K,L,N,LP1,KP,J,I DEFINED AS FIXED CAN NOT BE USED
*SINCE THEY ARE PART OF THE MAIN SIMULATION
* EXPLANATION OF THE THEORY FOR THE PRESENT PROGRAM CAN BE FOUND IN
* E.ROMERO THESIS JUN/72
FIXED M,M2,M1,K,L,N,LP1,KP,J,I
*SECTION 1 -COEFFICIENTS FOR A PARTICULAR SHIP
*INTRODUCE: TITLE,DEFLECTIONS,HYDRODYNAMIC COEFFICIENTS AS CSMP PARAMETERS
*SECTION 2 -PARAMETERS CALCULATIONS
*AFTER REARRANGING THE EQUATIONS IN THE FORM SHOWN AT THE BEGINNING OF THIS
*PROGRAM,DEFINE COEFFICIENTS OF POLYNOMIAL IN 'S' IN FUNCTION OF THE
```

```
*HYDRODYNAMIC COEFFICIENTS.IN SECTION 2A THE COEFFICIENTS OF THE HIGHEST
*ORDER TERM IN 'S' ARE INTRODUCED AS INDICATED.AFTER 'DYNAMIC' THE REST OF
*THE COEFFICIENTS ARE INTRODUCED,NO ORDER IS REQUIRED.
INITIAL
*SECTION 2A -ALL PARAMETERS MUST BE DEFINED AND IN SEQUENCE:
*AAA,AAB,AAC,AAD,AAE,AAF,ABA,ABB,ABC,ABD,ABE,ABF,ACA,ACB,ACC,ACD,ACE
*ACF,ADA,ADB,ADC,ADD,ADE,ADF,AEA,AEB,AEC,AED,AEE,AEF,AFA,AFB,AFC,AFD,AFE,AFF
*END SECTION 2A
      DEL=VALUE(AAA,0,0)
      COFAA=VALUE(AAA,1,1)
      COFAB=VALUE(AAA,2,1)
      COFAC=VALUE(AAA,3,1)
      COFAD=VALUE(AAA,4,1)
      COFAE=VALUE(AAA,5,1)
      COFAF=VALUE(AAA,6,1)
      COFBA=VALUE(AAA,1,2)
      COFBB=VALUE(AAA,2,2)
      COFBC=VALUE(AAA,3,2)
      COFBD=VALUE(AAA,4,2)
      COFBE=VALUE(AAA,5,2)
      COFBF=VALUE(AAA,6,2)
      COFCA=VALUE(AAA,1,3)
      COFCB=VALUE(AAA,2,3)
      COFCC=VALUE(AAA,3,3)
      COFCD=VALUE(AAA,4,3)
      COFCE=VALUE(AAA,5,3)
      COFCF=VALUE(AAA,6,3)
      COFDA=VALUE(AAA,1,4)
      COFDB=VALUE(AAA,2,4)
      COFDC=VALUE(AAA,3,4)
      COFDD=VALUE(AAA,4,4)
      COFDE=VALUE(AAA,5,4)
      COFDF=VALUE(AAA,6,4)
      COFEA=VALUE(AAA,1,5)
      COFEB=VALUE(AAA,2,5)
      COFEC=VALUE(AAA,3,5)
      COFED=VALUE(AAA,4,5)
      COFEE=VALUE(AAA,5,5)
      COFEF=VALUE(AAA,6,5)
      COFFA=VALUE(AAA,1,6)
      COFFB=VALUE(AAA,2,6)
      COFFC=VALUE(AAA,3,6)
      COFFD=VALUE(AAA,4,6)
      COFFE=VALUE(AAA,5,6)
      COFFF=VALUE(AAA,6,6)
DYNAMIC
*INTRODUCE REST OF COEFFICIENTS.NO ORDER REQUIRED,NON DEFINED TERMS ARE SET=0.
*SECTION 3-DEFINITIONS
```

```
*INTRODUCE DEFINITIONS OF THE VARIABLES OBTAINED FROM COMPARISON WITH THE
* EQUATIONS AT THE BEGINNING OF THE PROGRAM AND THE REAL MEANING,EG:YAW=B.
*INTRODUCE THE NONLINEAR TERMS IN THE CORRESPONDING EQUATION BY DEFINING
*NA,NB,NC,ND,NE,NF . WHEN THEY ARE NOT DEFINED ARE SET AUTOMATICALLY TO ZERO
*INTRODUCE ANY OTHER DEFINITION EG.:KINEMATIC RELATIONS.USE OF SUBPROGRAMS
*IS POSSIBLE.
*SECTION 4 -PROGRAMMED SIMULATION
      I1=-BAA*ADOT-GAA*A-BBA*BDOT-GBA*B-BCA*CDOT-GCA*C...
      -BDA*DDOT-GDA*D-BEA*EDOT-GEA*E-BFA*FDOT-GFA*F+IF1
      I2=-BAB*ADOT-GAB*A-BBB*BDOT-GBB*B-BCB*CDOT-GCB*C...
      -BDB*DDOT-GDB*D-BEB*EDOT-GEB*E-BFB*FDOT-GFB*F+IF2
      I3=-BAC*ADOT-GAC*A-BBC*BDOT-GBC*B-BCC*CDOT-GCC*C...
      -BDC*DDOT-GDC*D-BEC*EDOT-GEC*E-BFC*FDOT-GFC*F+IF3
      I4=-BAD*ADOT-GAD*A-BBD*BDOT-GBD*B-BCD*CDOT-GCD*C...
      -BDD*DDOT-GDD*D-BED*EDOT-GED*E-BFD*FDOT-GFD*F+IF4
      I5=-BAE*ADOT-GAE*A-BBE*BDOT-GBE*B-BCE*CDOT-GCE*C...
      -BDE*DDOT-GDE*D-BEE*EDOT-GEE*E-BFE*FDOT-GFE*F+IF5
      I6=-BAF*ADOT-GAF*A-BBF*BDOT-GBF*B-BCF*CDOT-GCF*C...
      -BDF*DDOT-GDF*D-BEF*EDOT-GEF*E-BFF*FDOT-GFF*F+IF6
      IF1=KA1*D1+KA2*D2+NA
      IF2=KB1*D1+KB2*D2+NB
      IF3=KC1*D1+KC2*D2+NC
      IF4=KD1*D1+KD2*D2+ND
      IF5=KE1*D1+KE2*D2+NE
      IF6=KF1*D1+KF2*D2+NF
      ADDOT=(COFAA*I1+COFAB*I2+COFAC*I3+COFAD*I4+COFAE*I5+COFAF*I6)/DEL
      BDDOT=(COFBA*I1+COFBB*I2+COFBC*I3+COFBD*I4+COFBE*I5+COFBF*I6)/DEL
      CDDOT=(COFCA*I1+COFCB*I2+COFCC*I3+COFCD*I4+COFCE*I5+COFCF*I6)/DEL
      DDDOT=(COFDA*I1+COFDB*I2+COFDC*I3+COFDD*I4+COFDE*I5+COFDF*I6)/DEL
      EDDOT=(COFEA*I1+COFEB*I2+COFEC*I3+COFED*I4+COFEE*I5+COFEF*I6)/DEL
      FDDOT=(COFFA*I1+COFFB*I2+COFFC*I3+COFFD*I4+COFFE*I5+COFFF*I6)/DEL
      ADOT=INTGRL(ADOTO,ADDOT)
      BDOT=INTGRL(0.,BDDOT)
      CDOT=INTGRL(0.,CDDOT)
      DDOT=INTGRL(0.,DDDOT)
      EDOT=INTGRL(0.,EDDOT)
      FDOT=INTGRL(0.,FDDOT)
      A=INTGRL(0.,ADOT)
      B=INTGRL(0.,BDOT)
      C=INTGRL(0.,CDOT)
      D=INTGRL(0.,DDOT)
      E=INTGRL(0.,EDOT)
      F=INTGRL(0.,FDOT)
*SECTION 5 -OUTPUT CHARACTERISTICS
*FOLLOW CSMP RULES TO INTRODUCE TIME, OUTPUT DESIRED AND LABEL
END
STOP
```

```
      FUNCTION VALUE(Y,I,M)
COMMON
      DIMENSION X(6,6),Y(6,6)
      DO 1 M1=1,6
      DO 1 M2=1,6
    1  X(M1,M2)=Y(M1,M2)
      IF(I.EQ.0) GO TO 100
      X(I,1)=0.
      X(I,2)=0.
      X(I,3)=0.
      X(I,4)=0.
      X(I,5)=0.
      X(I,6)=0.
      X(1,M)=0.
      X(2,M)=0.
      X(3,M)=0.
      X(4,M)=0.
      X(5,M)=0.
      X(6,M)=0.
      X(I,M)=1.
  100 CONTINUE
      WRITE(6,49) I,M
   49 FORMAT(/,' ','DETERMINANT FOR COF',I1,I1,':')
      DO 50 M1=1,6
      WRITE(6,51)(X(M1,M2),M2=1,6)
   51 FORMAT(10(1X,E13.6))
   50 CONTINUE
      N=6
      DD=1.00
   10  DO 34 L=1,N
       KP=0
       Z=0.0
       DO 12 K=L,N
      IF(Z-ABS(X(K,L)))11,12,12
   11 Z=ABS(X(K,L))
       KP=K
   12  CONTINUE
       IF(L-KP)13,20,20
   13  DO 14 J=L,N
       Z=X(L,J)
       X(L,J)=X(KP,J)
   14  X(KP,J)=Z
      DD=-DD
   20  IF(L-N)31,40,40
   31  LP1=L+1
       DO 34 K=LP1,N
       IF(X(K,L))32,34,32
   32  RATIO=X(K,L)/X(L,L)
```

```
      DO 33 J=LP1,N
   33 X(K,J)=X(K,J)-RATIO*X(L,J)
   34 CONTINUE
   40 DO 41 K=1,N
   41 DD=DD*X(K,K)
      D=DD
      VALUE=D
      WRITE(6,52) I,M,VALUE
   52 FORMAT(' ','COF',I1,I1,'=',E15.6)
      RETURN
      END
ENDJOB
```

COMPUTER PROGRAM 7

SUBROUTINE TO CALL DRAW

```
// EXEC FORTCLGP,COND.LINK=(4,LT,FORT),
// COND.GO=((4,LT,FORT),(4,LT,LINK)),
// REGION.GO=100K
//FORT.SYSIN DD *
      DIMENSION XOUT(50),X(900),Y(900),TYPE(2)
      REAL*8  LABEL/8H        /,ITITLE(12),BLANK/'        '/,TGRL,VAR1,
     1VAR2,VAR3,VAR4,VAR5,VAR6,TEST
      DATA ENDS/'ENDS'/
      DOUBLE PRECISION WORD
C
C     KGRAPH IS THE NUMBER OF OUTPUT VARIABLES
      KGRAPH=2
C
CC    READ USER-SUPPLIED CONSTANTS FOR DRAW
C
      READ(5,7) NOPLOT
   16 READ(5,7) NOCUR
    7 FORMAT(I3)
      READ(5,1) (ITITLE(I),I=1,6)
      READ(5,1) (ITITLE(I),I=7,12)
    1 FORMAT(6A8)
      READ(5,2)  XSCALE,YSCALE,IUP,IRITE,MODEX,MODEY,IW,IHI,IGR
    2 FORMAT(2F5.0,7I5)
      IF(NOCUR-1)  14,10,11
   10 MODCUR=0
      GO TO 15
   11 MODCUR=1
C
C
C     READ DATA SET OF POINTS PRODUCED BY CSMP
C
   15 CONTINUE
C
C     READ 10 INTRODUCTORY RECORDS
C
      DO 3 I=1,10
    3 READ  (15) WORD
C
C     READ LABEL CORRESPONDING TO END OR CONTINUE
C
      READ (15) WORD
      NUMPTS=0
C
```

```
C      READ POINTS
C
     6 READ (15) (XOUT(I),I=1,KGRAPH)
       IF (XOUT(1) - ENDS) 8,9,8
     8 NUMPTS = NUMPTS + 1
       X(NUMPTS)=XOUT(1)
       Y(NUMPTS)=XOUT(2)
       GO TO 6
     9 CONTINUE
       DO 18 I=1,NUMPTS
       WRITE (6,19) X(I),Y(I)
    19 FORMAT (5X,F20.4,10X,F20.4)
    18 CONTINUE
C
C      CALL DRAW ROUTINE
C
       CALL DRAW(NUMPTS,X,Y,MODCUR,0,LABEL,ITITLE,XSCALE,YSCALE,IUP,IRITE
      1,MODEX,MODEY,IW,IHI,IGR,LAST)
       NOCUR = NOCUR - 1
       IF(NOCUR - 1) 14,12,13
    12 MODCUR = 3
       READ (15) WORD
       GO TO 15
    13 MODCUR = 2
       READ (15) WORD
       GO TO 15
    14 NOPLOT = NOPLOT - 1
       IF(NOPLOT)  17,17,16
    17 STOP
       END
//GO.FT15F001 DD DSNAME=&PLOTPE,UNIT=SYSDA,DISP=(OLD,DELETE)
//GO.SYSIN DD *
001
001
TITLE
TITLE
0    0        4    4    0    0    8    5    1
```

LIST OF REFERENCES

1. Abkowitz, Martin A., Stability and Motion Control of Ocean Vehicles, the MIT Press, 1969.

2. Adair, Jamie and Gillmer, T., Fundamentals of Naval Construction and Ship Stability, The United States Naval Institute, Annapolis, Maryland, 1953.

3. Gertler, M. and Hagen, G., Standard Equations of Motion for Submarine Simulation, Naval Ship Research and Development Center, Report 2510, June 1967.

4. Higdon, D. T., Dynamics and Control of a Twin-Hulled Semisubmerged Ship (S3), Naval Undersea Research and Development Center, San Diego, California, August 1971.

5. Hsu, J., and Meyer, A., Modern Control Principles and Applications, McGraw-Hill Book Company, 1968.

6. Johnson, C., Analog Computer Techniques, McGraw-Hill Book Company, 1963.

7. McCain, J. H., Computer Program Describing Submarine Motion with Six Degrees of Freedom, Naval Ship Research and Development Center, Report, July 1967.

8. Merzhauser, Klaus P., Symbolic Algebraic Manipulation by Digital Computer in Problems of Control Theory, Thesis, Naval Postgraduate School, Monterey, California, 1970.

9. Rekoff, M., Analog Computer Programming, Charles E. Meril Books, Inc., 1967.

10. Thaler, G., Borthwick, R., and McCamey, R., Self-Adaptive Control Systems, Technical Report No. 20, Naval Postgraduate School, Monterey, California, 1961.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Dr. G. J. Thaler, Code 52 Tr Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	10
4.	LTjg Edgar Romero 3rd Avenida - Manga No. 2106 Cartagena Colombia, South America	5
5.	Mr. D. R. Towill University of Wales Institute of Science and Technology Cathays Park, Cardiff, Wales	1
6.	Professor M. L. Wilcox, Code 52 Wx Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
7.	Professor A. Gerba, Jr., Code 52 Gz Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
8.	Professor D. M. Layton, Code 57 Ln Aeronautical Engineering Department Naval Postgraduate School Monterey, California 93940	1
9.	Mr. Walter Blumberg NSRDC Annapolis, Maryland	1
10.	Mr. Pat Patterson NSRDC Annapolis, Maryland	1

11. Dr. Grant Hagen — 1
NSRDC Carderock Laboratory
Bethesda, Maryland 20034

12. Mr. J. Russ — 1
NSRDC Carderock Laboratory
Bethesda, Maryland 20034

13. Sr. Vicealmirante — 2
Comandante Armada Nacional
Ministerio de Defensa
Bogota, Colombia
South America

14. Sr. Contralmirante — 2
Director Escuela Naval de Cadetes
Manzanillo, Cartagena
Colombia, South America

15. Sr. Capitan de Navio — 1
Guillermo Uribe P.
Commandante ARC Gloria
Base Naval ARC Bolivar
Cartagena, Colombia
South America

16. Sr. Teniente de Navio — 1
Jorge Cadena
Direccion de Personal
Comando Armada Nacional
Bogota, Colombia
South America

17. Profesor Ing. — 1
Luis Polo
Escuela Naval de Cadetes
Manzanillo, Cartagena
Colombia, South America

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

ORIGINATING ACTIVITY (Corporate author)

Naval Postgraduate School
Monterey, California 93940

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

REPORT TITLE

Mathematical Models and Computer Solution for the Equations of Motion of Surface Ships and Submarines, in Six Degrees of Freedom

DESCRIPTIVE NOTES (Type of report and inclusive dates)

Master's Thesis; June 1972

AUTHOR(S) (First name, middle initial, last name)

Edgar Romero

REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1972	134	10

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

From the standard equations of motion, mathematical models are developed and standardized for simulation in analog or digital computers. A complete study is done of simulation techniques as applied to the mathematical models. A new technique for simulation of very complex systems of equations including nonlinearities is developed. The model can be used in control studies. Digital simulations of the dynamics of the semisubmerged ship and a submarine in six degrees of freedom are done. This latter with all the nonlinearities and cross-coupling terms included.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

Security Classification

A-31408

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
ship control						
ship dynamics						
simulation						
mathematical models in six degrees of freedom						
ship maneuvering						

Thesis
R6897 Romero
c.1

135501

Mathematical models and computer solution for the equations of motion of surface ships and submarines, in six degrees of freedom.

14 APR 76
18 AUG 78
12 Sep 79
23 FEB 89

22525
23200
24773
S12366
27784
32351

Thesis
R6897 Romero
c.1

135501

Mathematical models and computer solution for the equations of motion of surface ships and submarines, in six degrees of freedom.

thesR6897
Mathematical models and computer solutio
3 2768 001 00295 9
DUDLEY KNOX LIBRARY